



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **07295929 A**(43) Date of publication of application: **10.11.95**

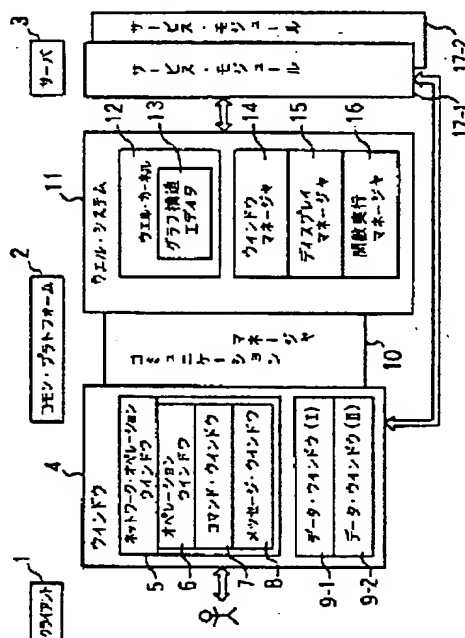
(51) Int. Cl.

G06F 15/00**G06F 15/16**(21) Application number: **07041572**(22) Date of filing: **01.03.95**(30) Priority: **04.03.94 JP 06 34701**(71) Applicant: **FUJITSU LTD**(72) Inventor: **ENOMOTO HAJIME****(54) INTERACTIVE INFORMATION PROCESSOR BY
COMMON PLATFORM FUNCTION****(57) Abstract:**

PURPOSE: To effectively advance interaction by matching a data notation form or the like between a client and a server by a common platform provided with the various kinds of windows.

CONSTITUTION: The server 3 calls an object network defined as a processing object by the client 1 and a graph structure editor 13 stores it in a well kernel 12. Then, the object network is displayed in an operation window 6 through a communication manager 10. In the state, the client 1 specifies the displayed network and indicates its own intention. Then, the manager 10 calls a template relating to the object at a processing object position and displays the constraint data in a data window 9-2. The client 1 selects the constraint data, the server 3 executes an execution schedule in service modules 17-1 and 17-2 and the result is displayed in the data window 9-1.

COPYRIGHT: (C)1995,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-295929

(43) 公開日 平成7年(1995)11月10日

(51) Int.Cl.⁵

G 0 6 F 15/00
15/16

識別記号

3 1 0 R 7459-5L
3 7 0 N

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数21 O L (全 26 頁)

(21) 出願番号 特願平7-41572

(22) 出願日 平成7年(1995)3月1日

(31) 優先権主張番号 特願平6-34701

(32) 優先日 平6(1994)3月4日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72) 発明者 榎本 肇

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74) 代理人 弁理士 森田 寛 (外2名)

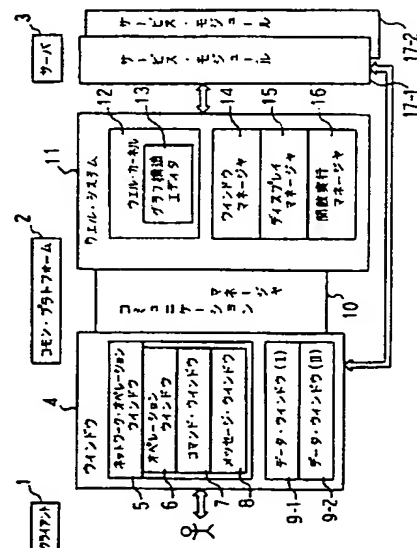
(54) 【発明の名称】 コモン・プラットフォーム機能による対話的情報処理装置

(57) 【要約】

【目的】 本発明は、各種ウインドウをもつコモン・プラットフォームを用意し、クライアントとサーバとの対話を効果的に進めることを目的とする。

【構成】 クライアントとサーバとの間にもうけたコモン・プラットフォームにおいて、クライアントとサーバとの間でのデータ表記1式などの整合をとるコミュニケーション・マネージャをもうけ、システム内でテンプレートの形で与えられるデータを取り扱うようにしている。

本発明の原理構成図



【特許請求の範囲】

【請求項1】 オブジェクト・ネットワークに対応した処理を実行する1つまたは複数のサービス・モジュールがサーバとして用意され、

クライアントの要求する意図に応じて、上記サーバが情報処理を実行する情報処理装置において、

上記クライアント（1）と上記サーバ（3）との間の対話の場としてコモン・プラットフォーム（2）が設定されており、

当該コモン・プラットフォーム（2）が、
上記オブジェクト・ネットワーク（19）に対応した可視化対象項目を表示し、かつ上記クライアント（1）が意図する指示とデータとを表示すると共に、上記サーバ（3）が実行した結果を表示するウインドウ（4）と、
上記ウインドウ（4）上の可視化対象項目を選択する処理を行うウインドウ・マネージャ（14）と、上記ウインドウ（4）上に可視化対象項目を表示する処理を行うディスプレイ・マネージャ（15）と、上記クライアント（1）の意図に対応した処理を実行するための関数を実行するための関数実行マネージャ（16）と、オブジェクト・ネットワーク（19）を管理するグラフ構造エディタ（13）と、処理を実行することを制御するためのウエル・カーネル（12）とを少なくともも有するウエル・システム（11）と、

上記クライアント（1）の意図する所を上記サーバ（3）に伝え、かつ上記サーバからの処理結果を上記ウインドウ（4）に対して可視的に表示する仲介を行うコミュニケーション・マネージャ（10）とを少なくとももそなえたことを特徴とするコモン・プラットフォーム機能による対話的情報処理装置。

【請求項2】 クライアント（1）の要求する意図について実行すべきオブジェクトの構造としてオブジェクト・ネットワーク（19）を上記ウインドウ（4）上に表示し、

上記クライアント（1）が上記表示されたオブジェクト・ネットワーク（19）について操作を行うよう構成されており、

上記コミュニケーション・マネージャ（10）は、上記クライアント（1）による操作結果を上記サーバ（3）のもつ言語処理機能に伝え、かつ上記サーバ（3）による処理結果を上記ウインドウ（4）上に可視的に表示せしめるよう構成されて、上記クライアント（1）と上記サーバ（3）との間のデータ表記形式の整合をとることを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項3】 上記オブジェクト・ネットワーク（19）を構成する個々のオブジェクトに対応して、当該オブジェクトを特定しかつ当該オブジェクトに関する属性を記述したテンプレート（22）が用意されるよう構成されており、

1つのオブジェクトに対応するテンプレート（22）中の属性に当該1つのオブジェクトに対して修飾を行ったことを表す修飾語を記述することによって、新しいテンプレートが生成され、当該新しいテンプレートが当該1つのオブジェクトから派生されたオブジェクトを特定することを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項4】 上記オブジェクト・ネットワーク（19）を構成する個々のオブジェクトに対応して、当該オブジェクトを特定しかつ当該オブジェクトに関する属性を記述したテンプレート（22）が用意されるよう構成されており、

かつ上記クライアント（1）が上記オブジェクト・ネットワークに対して要求する意図に対応して、上記テンプレートの組み合わせが用意されると共に、

上記コミュニケーション・マネージャ（10）が介在して、上記テンプレートの内容を上記サーバ（3）と上記クライアント（1）との間のデータ表記形式の整合をとるようにしたことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項5】 上記クライアント（1）が上記オブジェクト・ネットワークに対して要求する意図に対応して、上記コミュニケーション・マネージャ（10）が、上記ネットワーク中の部分的あるいは全体的オブジェクト構造の特定部分を指示しその部分をグラフ構造エディタを動作させて、オブジェクト・ネットワークの内容を書き替え可能となるよう記述し、

上記コミュニケーション・マネージャ（10）が、当該記述された結果のテンプレートにもとづいて、上記ウインドウ（4）上のオブジェクト・ネットワークの構造に反映するようにしたことを特徴とする請求項4記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項6】 上記サーバ（3）の1つがエイジェント機能をもつよう構成され、

当該エイジェント機能をもつサーバ（3'）が、第1の上記コモン・プラットフォーム（2-1）を介して指示された上記クライアント（1）の意図を詳細化し当該意図に対して対応可能なサーバ（3''）の群をファイルの中から選択し、

当該エイジェント機能をもつサーバ（3'）が、上記クライアント（1）に代わって、第2の上記コモン・プラットフォーム（2-2）におけるクライアントの役割を実行し、

第2の上記コモン・プラットフォーム（2-2）に対して、上記クライアント（1）の意図を仲介翻訳し、上記詳細化された意図を通知し、

第2の上記コモン・プラットフォーム（2-2）を介して上記サーバ（3''）からの処理結果を受取り、

第1の上記コモン・プラットフォーム（2-1）を介し

て、上記クライアント(1)に対して、上記サーバ(3'')による処理結果を通知するようにしたことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項7】 上記オブジェクト・ネットワークは、パラメータおよび/または制約条件を一般化されて与えられているジェネリック・オブジェクト・ネットワーク(33)と、当該ジェネリック・オブジェクト・ネットワーク(33)に対して特定の処理のために必要なパラメータおよび/または制約条件を与えられて構成される10 スペシフィック・オブジェクト・ネットワーク(35)との2階層をもつよう構成されることを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項8】 上記オブジェクト・ネットワークは、パラメータおよび/または制約条件を一般化されて与えられているジェネリック・オブジェクト・ネットワーク(33)と、当該ジェネリック・オブジェクト・ネットワーク(33)に対して特定の処理のために必要なパラメータおよび/または制約条件を与えられて構成される20 スペシフィック・オブジェクト・ネットワーク(35)との2階層をもつよう構成され、

上記エイジェント機能をもつサーバ(3')は、上記クライアント(1)が意図する指示に対応して、当該意図を実行するに足る上記ジェネリック・オブジェクト・ネットワークを検索し、かつ当該意図を実行するよう上記パラメータおよび/または制約条件を与えられた上記スペシフィック・オブジェクト・ネットワークとを選定し、上記指示されたクライアント(1)の意図を詳細化し当該意図に対して対応可能な上記サーバ(3'')の群を選択することを特徴とする請求項8記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項9】 上記エイジェント機能をもつサーバ(3')は、上記検索されて得られるジェネリック・オブジェクト・ネットワークと上記選定されたスペシフィック・オブジェクト・ネットワークとに対応する上記サーバ(3'')の群についての整合をとる機能と、個々のサーバ(3'')に対応する個別サービスの実行に対して規定される制約条件と当該個々のサーバ(3'')を組合わせた統合サービスの実行に対して規定される制約条件とを整合させる機能とを少なくとも有するようにしたことを特徴とする請求項8記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項10】 上記ウィンドウ(4)は、上記オブジェクト・ネットワークを表示するオペレーション・ウィンドウ(6)と、データの入出力を行うデータ・ウィンドウ(9)と、上記サーバ(3)と上記クライアント(1)との間でのメッセージを表示するメッセージ・ウィンドウ(8)

と、上記サーバ(3)に対するコマンドを表示するコマンド・ウィンドウ(7)と、を少なくとも有し、各ウィンドウ(6, 7, 8, 9)が選択的にあるいは並存的に表示されることを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項11】 オブジェクト・ネットワーク(19)中のある名詞オブジェクト(27)から隣接ノードとして存在する他の名詞オブジェクト(27)への推移を動詞オブジェクトの実行に対応づけておくよう構成しており、上記他の名詞オブジェクトに対応するテンプレート(22)から、および動詞オブジェクトに対応するテンプレート(22)から、上記推移の結果に対応する属性を抽出するよう構成し、上記推移の結果をチェックするために入力した制約条件と対比するようにしたことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項12】 上記コミュニケーション・マネージャ(10)は、上記クライアント(1)が上記ウィンドウ(4)において操作要求を行うイベント型の駆動と、上記ウィンドウ(4)において上記クライアント(1)に対してデータを要求するデータ型の駆動とを管理するよう構成されてなり、上記コミュニケーション・マネージャ(10)は、上記イベント型の駆動に対応して、当該イベント型の駆動で要求される処理に対してデータが不足していることを調べる機能をもち、当該データの不足に対応して上記データ型の駆動を行うようにしたことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項13】 上記コミュニケーション・マネージャ(10)は、オブジェクト・ネットワークの現在状態から、並行実行可能な動詞オブジェクトを検出し、上記イベント型の駆動指令を発して自ら並行実行を行わせるようにしたことを特徴とする請求項12記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項14】 上記コミュニケーション・マネージャ(10)は、上記ジェネリック・オブジェクト・ネットワークで規定される処理サービスを実行すべく上記スペシフィック・ロール・サーバに対して実行命令を与えるに当たって、当該命令を実行する個々のスペシフィック・ロール・サーバに与える実行命令の群中の各命令を時相的制約条件を満足するように発生させる機能を、上記ウエル・システム(11)内に持つよう構成され、かつ上記コミュニケーション・マネージャ(10)は、当該各命令の発生状況を上記コモン・プラットフォーム(2)を介してクライアント(1)に提示するようにし

たことを特徴とする請求項7記載のコモン・プラットフォーム機能による対話的情報処理装置

【請求項15】 上記コミュニケーション・マネージャ(10)は、複数の上記サーバが実行する上での制約条件の群を、上記メッセージ・ウィンドウを介して表示し、かつ当該制約条件の群および上記クライアントから入力された制約条件の群の相互の間での妥当性をチェックする機能をもつようにしたことを特徴とする請求項8記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項16】 上記エイジェント・ロール・サーバは、上記ジェネリック・オブジェクト・ネットワークと上記スペシフィック・オブジェクト・ネットワークとによる処理の進行の状況を、上記コモン・プラットフォームを介してクライアントに表示するようにしたことを特徴とする請求項8記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項17】 上記コモン・プラットフォームを介して表示された情報にもとづいて、上記クライアントが当該情報を対話的に修正するようにしたことを特徴とする請求項16記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項18】 上記スペシフィック・ロール・サーバにおいては、オブジェクトの属性についての継承と、当該属性についての他オブジェクトの属性との合成とについて、当該サーバに付属しているコモン・プラットフォームを支援するコミュニケーション・マネージャが管理するよう構成されており、上記合成が行われる際の合成データは、当該スペシフィック・ロール・サーバが上位階層のサーバに対して転送するよう構成されていることを特徴とする請求項8記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項19】 上記コモン・プラットフォーム(2-1)は上記サーバに対して処理の実行依頼するユーザ(1-1)に対するウィンドウをもつと共に、当該ユーザ(1-1)の意図に対応して当該ユーザ(1-1)が行う情報の授受態様を支援するエイジェント・エキスパート(1-2)がもうけられ、当該エイジェント・エキスパート(1-2)は、グラフ構造エディタを用いて、ユーザ(1-1)に対する上記ウィンドウ中のジェネリック・オブジェクト・ネットワークを設定および/または改訂を可能とするよう記述し、その後ユーザ(1-1)に当該オブジェクト・ネットワークを使用できる旨を通知する機能をもつことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項20】 上記エイジェント・ロール・サーバ(3')と上記スペシフィック・ロール・サーバに属する夫々のサーバとの間に介在するコモン・プラットフォ

ーム(2-2)は、上記エイジェント・ロール・サーバ(3')に対してウィンドウをもつと共に、当該エイジェント・ロール・サーバ(3')の意図に対応して当該エイジェント・ロール・サーバ(3')が行う情報の授受態様を支援するスペシフィック・エキスパート(1-3)に対するウィンドウがもうけられ、当該スペシフィック・エキスパート(1-3)は、グラフ構造エディタを用いて、スペシフィック・オブジェクト・ネットワークを設定および/または改訂する機能をもつことを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【請求項21】 名詞オブジェクトに対して修飾を行った形容詞に対応して、当該形容詞の属性構造を集積し、当該集積された属性構造を、上記名詞オブジェクトに対するテンプレートに付加したテンプレートが、当該形容詞に対するテンプレートとして作成されることを特徴とする請求項1記載のコモン・プラットフォーム機能による対話的情報処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、クライアントがサーバと対話を行いながら、クライアントの意図にしたがったサービスを実行してゆく場としての各種ウィンドウをもつコモン・プラットフォームを用意し、コモン・プラットフォーム内のコミュニケーション・マネージャが上記ウィンドウを介在させて上記クライアントと上記サーバとの間でのデータ表記形式などの整合をとるなどの仲介を行うようにしたコモン・プラットフォーム機能による対話的情報処理装置に関する。

【0002】

【従来の技術】 従来から情報処理装置の画面に向かって対話を行いつつ処理を実行することが行われる。

【0003】 ある課題を解決し処理結果を得るためには、好奇心から意図が発生し、その意図を満足させるようなサービス・システムが必要となる。このために図30のようなクライアント・サーバ間の対話モデルを考えると、意図をグラフ構造によって表現したオブジェクト・ネットワークによって定義することが考えられる。そして、個々の意図が逐次あるいは並行的にサービスを行う対話図式によって、オブジェクト・ネットワークの内容が表現される。

【0004】 図30の左側はクライアントであって当該クライアントが行う意図についての内容記述作業を矩形の枠で示し、図示の右側はサーバであって当該サーバが行う処理を矩形の枠で示し、中央部に点線で囲ってディスプレイなどの表示機能を含む表示のプロセスが介在していることを表している。

【0005】 クライアントは興味をもったサービス名によって意図として、表示のプロセスにサービスの要求を行う①。サーバは当該要求に対応して意図実現計画を実

10

20

30

40

50

行して当該処理の対象となるオブジェクト・ネットワークを呈示する②。

【0006】当該呈示されたオブジェクト・ネットワークを画面でみながら、クライアントは単位意図の選択を行う③。この選択された単位意図はサーバ側に示されてそのときの制約データや条件や関数計画を実行して画面上で呈示する④。

【0007】クライアントは呈示されたデータや条件や関数を選択する⑤。サーバ側では選択されたデータや条件や関数にもとづいてサービス機能実行計画を実行してその実行結果を画面上で呈示する⑥。クライアントはその実行結果を評価して次の意図選択を行う⑦。

【0008】上記の如くクライアントおよびサーバが協調して意図を実現する。即ちクライアントは意図を選択する過程でサーバに単位意図を要求する。これに答えてサーバはサービス実行に必要な制約条項を計画し、クライアントに向けて、その必要性を応答する。このようなインタラクションのための場が対話に有効である。

【0009】対話が円滑であるためには、対話プロセス意味論上の結束性が、意図駆動による意図内容の接続的整合性によって表現され、それはオブジェクト・ネットワークの妥当性を保証する。

【0010】

【発明が解決しようとする課題】対話プロセスを円滑に進行させるためには、サービス・プロセスに役割を特化させるのが効率的である。人間にとって表現シンボルとしては、画像・図形の方がテキスト文よりも一眼で直観的に理解されるのに対し、計算機は論理的記述が処理し易い。役割は、理解解読機能、実行処理機能などの諸機能、あるいは述語や表現スタイルなどの記述形式によって規定される。

【0011】本発明は、各種ウインドウをもつ共通・プラットフォームを用意し、共通・プラットフォーム内のコミュニケーション・マネージャがクライアントとサーバとの間でのデータ表記形式などの整合をとるようにして、対話を効果的に進めうようにすることを目的としている。

【0012】

【課題を解決するための手段】図1は本発明の原理構成図を示す。図中の符号1はクライアント、2は共通・プラットフォーム、3はサーバを表している。共通・プラットフォームにおいて、4はディスプレイなどの表示手段上に表示されるウインドウであり、大別して、ネットワーク・オペレーション・ウインドウ5とデータ・ウインドウ9とに分けられる。そして、6はオペレーション・ウインドウであって各種オペレーションを指示可能にする画像や文字を表示するもの、7はコマンド・ウインドウであって各種コマンドを指示可能にする画像や文字を表示するもの、8はメッセージ・ウインドウであって各種メッセージを表示可能にするものを表してい

る。

【0013】またデータ・ウインドウ9は、処理結果を表すデータ・ウインドウ(I)9-1と、処理に必要な制約データなどを表すデータ・ウインドウ(II)9-2とからなる。

【0014】10はコミュニケーション・マネージャであって、ウインドウ4を介して交換されるクライアント1とサーバ3との交信を仲介する。11はウエル・システム(各種のオブジェクト・ネットワークに対処できる拡張性をもっているという意味で、以下、エキステンシブ・ウエル・システムと呼ぶことがある)である。12はウエル・カーネルであって上記コミュニケーション・マネージャ10や以下の各種マネージャや、グラフ構造エディタを管理し、また作業領域をもっている。13はグラフ構造エディタであってオブジェクト・ネットワークをも一種のデータと考えて、グラフ構造をテンプレートとして表現して処理を行う。

【0015】14はウインドウ・マネージャであって、ウインドウ4の選択などを処理する機能をもつものである。15はディスプレイ・マネージャであって、ウインドウを表示するディスプレイを制御する機能をもつものである。16は関数実行マネージャであって、オブジェクト・ネットワークに対してクライアントの意図に沿った処理を行う上での関数を実行する上での処理を行う機能をもつものである。

【0016】17-1、17-2は夫々の目的に対応した処理を行うサービス・モジュールである。

【0017】

【作用】サーバ3はクライアントが処理対象にしようとしている意図内容を記述しているオブジェクト・ネットワークを呼び出して、グラフ構造エディタ13はウエル・カーネル12がもつ作業領域上に当該オブジェクト・ネットワークを記憶する。

【0018】この結果にもとづいて、ウインドウ・マネージャ14などの働きによってかつコミュニケーション・マネージャ10の仲介によってオブジェクト・ネットワークがオペレーション・ウインドウ6に表示される。

【0019】この状態の下で、クライアントは、表示されたオブジェクト・ネットワークにもとづいて、当該オブジェクト・ネットワーク上の全部または一部を特定して自己の意図を指示する。この指示に対応して、コミュニケーション・マネージャ10が当該意図を解釈し、サーバに対して処理対象となる位置に該当するオブジェクト(名詞オブジェクト)についてのテンプレートを呼び出させる。そしてサービス・モジュールが当該テンプレートに対応した領域準備などを行い、ウインドウ4はそのときの制約データの存在をデータ・ウインドウ(II)9-2に表示する。

【0020】クライアントは、制約データをデータ・ウインドウ(II)9-2上で選択し、サーバが実行計画を

実行し、サービス・モジュール17が実行し、ウィンドウ4が実行結果をデータ・ウィンドウ(I)9-1に表示する。その結果をクライアントが評価し、次の指示を行う。

【0021】

【実施例】図30に示した如きクライアントとサーバとの間の対話をより効果的に実現させるためには、クライアントと個々の処理を行うサーバとの間に、エイジェントの役割を行うエイジェント・ロール・サーバを介在させることが有効な場合がある。

【0022】コモン・プラットフォーム機能は、オブジェクト・ネットワークを表示するウィンド55と、実行に伴って必要とするデータおよび実行結果のデータを可視画像として表示するデータ・ウィンド99-iとを持ち、これらを支援するためのコミュニケーション・マネージャ10などを有しており、これらの機能によりクライアント1としてのユーザに最も適した表示形式を使用可能にし、それをコモン・プラットフォーム2において、データ処理装置内部での処理用に変換するようにして、ユーザにとって使い易くしている。

【0023】一方、サーバ3にとっては、個々のデータ*

*などはインデックス形式のタグが付いたテキスト形式で処理する必要がある。このようなクライアント1とサーバ3とはそれぞれ特有の役割をもっているため、コミュニケーション・マネージャ10はそれらの役割に応じたデータ形式に変換する役割をもっている。このためのデータの型を規定するためにテンプレートで制御を行うようにする。

【0024】このテンプレートにもとづいて、変換が行われるが、この場合クライアント1とサーバ3との役割の差に対応して表記の形式が異なる。クライアントである人間にとっては、テキスト形式よりは、グラフのような図や画像の方が分かり易く、また指示も容易である。特に点や線についてはデータ・ウィンド上で直接あるいはマウスによって参照指示が行えることが望ましい。一方、サーバとしての計算機にとっては、点については(x,y)の座標値として数値化され、線については始点から終点まで通過する画素点をリストの形式となっていなければ処理効率が悪い。このような観点から表1に示す如く役割機能に応じて表記形式が異なる必要がある。

【0025】

【表1】

テンプレート構造の役割に応じた表記形式

	クライアント	サーバ
点/線データ	点や線を実体で表示する (実体表示) 実体を参照しつつ指示する (参照指示)	インデックス形式で特定する コモン・プラットフォームへ一括転送し/連合処理する
図形/画像データ	画像や図形を実体で表示する 参照指示	リスト構造/ラスタ構造で特定する
データ要素	名前を指示する	名前ヘッダで特定する
サービス機能 (エージェント・サービス)	サービス構造を先ずジェネリック・オブジェクト・ネットワーク形式で与える サービス途中ではサービス項目サービス制約項目をもつものやサービス名を与える	サービス構造表を与える サービスを実行するサーバやサービスに当たって責任をもつサーバを指定する サービスを実行するサーバやサービスに当たって責任をもつサーバを指定する

【0026】即ち、点や線を表すデータについては、クライアント1との間では、点や線を実体で表示したクライアント1がそれぞれの実体を参照しつつ指示できるようにし、サーバ3との間では、インデックス形式で特定した特定されたものについて一括転送したり連合して処理したりする。

【0027】図形や画像を表すデータについては、クライアント1との間では、画像や図形を実体表示したクライアント1が参照指示できるようにし、サーバ3との間ではリスト構造やラスタ構造で特定するようにする。

【0028】データ要素については、クライアント1と

の間では名前を指示し、サーバ3との間では名前ヘッダで特定するようにする。サービス機能(エイジェントの抽象化サービス機能)については、クライアント1との間では、サービスをどのようなプログラムで行うかを明細にするサービス構造が先ずジェネリック・オブジェクト・ネットワーク形式で与えるようにする(クライアント1はこれにもとづいて所望するパラメータを与える)。そしてサービス実行の途中では、(i)サービス制約項目をパラメータとしてもつサービス項目や、(ii)サービス名をクライアント1が与え得るようにする。サーバ3との間では、(i)サービス構造表を与え、(ii)クラ

クライアント1からのサービス項目の指示に応じて、サービスを実行するサーバやサービスに当たって責任をもつサーバを指定するようにする。

【0029】図2はエイジェントをもつ構成を示している。図2に示すクライアント(Client)とサーバ(Specific role server)とが、図3.0に示すクライアント1とサーバ3' に対応している。そして、図2に示す場合には、エイジェント(Agent role server)が中間において、例えて言えば旅行仲介業者の如き役割をはたすサーバ3' としてもうけられるようにされる。

【0030】図2に示す表示的プロセス2-1および従属表示的プロセス2-2は夫々図3.0に示す表示的プロセスと同じ機能をもつものである。図2に示す場合には、クライアント1とエイジェント3' との間で表示的プロセス2-1に向かって、サービス要求と応答とが行われる。

【0031】エイジェント3' はクライアント1の意図にしたがったサービス計画を準備し、その役割を実行するサーバ(Specific role server)を検索し、サービス役割割当計画を行って、従属表示的プロセス2-2に向

かって役割機能を実行すべく要求する。

【0032】サーバ(Specific role server)3' は割当サービス実行計画を実行し、実行結果を従属表示的プロセス2-2に呈示する。エイジェント3' はサービス結果の内容をチェックして、サービス結果をクライアント1に呈示すべく表示的プロセス2-1に指示する。

【0033】図2に示す表示的プロセス2-1および従属表示的プロセス2-2は夫々、図1に示すコモン・プラットフォーム2によって実現される。そして図2に示すエイジェント3' は、図1に示すサービス・モジュール17-iの1つとして与えられると考えるべき。

【0034】図1に示すウエル・システム11は、WELL (Window-based Elaboration Language の略) と仮称される機能言語を取り扱う機能部であるとしているが、WELLと呼ばれる特別な言語を取り扱う機能部のみに限定して考えられるべきものではなく、ウインドウを用いて対話を行い得る、より広い意味の処理機能部が図1に示すウエル・システム11であると考えられるべきである。

【0035】図3はウエル・システムとオブジェクト・ネットワークとの関係を説明する図である。図中の符号11は図1に示すウエル・システムに対応している。また19A、19B、19Cは夫々図1に示すサービス・モジュールと組合わせられたオブジェクト・ネットワークであり、18A、18B、18Cは夫々所定の処理分野であり、図示の場合18Cはカラー画像生成・色付け処理分野を表している。

【0036】WELLと仮称される機能言語に対して分野に対応するオブジェクト・ネットワークを与えると当該オブジェクト・ネットワークの処理がプログラムレス

で実行が可能となる。また当該言語はウインドウ指向の言語であり、クライアントとのインターフェースはウインドウを介することによって、クライアント・サーバ・モデルが実現される。

【0037】図3においては、ウエル・システム11に対して、例えばカラー画像生成・色付け処理分野18Cに対応して必要とされるウインドウと当該対応する処理を行うサービス・モジュール定義とで与えられるオブジェクト・ネットワーク19Cを組合わせると、図示点線で示す如く当該ウエル・システムはカラー画像生成・色付け処理分野18Cに対応するシステムとなる。言うまでもなく他の分野のオブジェクト・ネットワーク19Aや19Bを組合わせると分野18Aや分野18Bに対応するシステムとなる。

【0038】システムを構成する言語処理系に与えられるオブジェクト・ネットワークを変更すれば、システム機能は変化する。処理系は、対象とするオブジェクト・ネットワークの構成記述を入力データとして与えることにより、該当する分野処理を行うのであるから、WELLは元来グラフ構造エディタ13を持っている。従ってシステムに、実行時にオブジェクト・ネットワークの変更機能を持たせることにより、システムの拡張性を持たせられる。特にインターフェースのために制約条件を名詞オブジェクトに持たせ、コモン・プラットフォーム中にあるメッセージ・ウインドウを利用できる。

【0039】互いに独立性のある多分野間の協調的統合のためにも拡張性は必要である。それは名詞オブジェクトについての制約関係の変更、動詞オブジェクトの実行制御についての同期性や排他性の動的変更など並行実行中のシステム環境への適応性がある。この場合、共通資財や計画修正などの管理の際に発生する役割転換を行う必要があり、得意機能に対応する対応処理機能を新しい共通の場の導入によって整合させる必要性が発生する。

【0040】図4は処理の整合を行う状況を概念的に示した図である。図示左端のクライアントが意図Aないし意図Nをもって対応するサービスAないしNを並行して実行させようとするとき、各サービスを実行する上での制約が存在することが多い。このような制約には、時相制約的なものと形態制約的なものとが存在し、前者によって実行時の規制やサービス間時相調整制御が行われ、後者によってサービス間形態調整制御が行われる。そして、このような規制や制御が行われた上で実行された複合サービス結果内容についてのチェックが行われ、クライアントに対してウインドウを介して呈示され、クライアントが評価を行うこととなる。

【0041】図5と図6とは一緒になって1つのフローチャートを示すものであって、図1に示す構成が用いられる態様を表すフローチャートを示す。

(S1): 例えばカラー画像生成・色付け処理分野18Cに対応する処理を行う場合、サーバが該当するオブジ

10

20

30

40

50

エクト・ネットワーク19Cを呼び出す。

【0042】(S2)：これに応じて、グラフ構造エディタは、ウエル・カーネル12上の作業領域上に当該オブジェクト・ネットワークを格納する。

(S3)：ウエル・カーネル12によって、図1示すウインドウ・マネージャ14やディスプレイ・マネージャ15を起動したカーネル12が当該各マネージャからの応答を受けるなどの処理をへて、コミュニケーション・マネージャ10の仲介によって、オブジェクト・ネットワークがオペレーション・ウインドウ6に表示される。

【0043】(S4)：クライアント1は、オペレーション・ウインドウ6上のオブジェクト・ネットワークをみた上で、当該オブジェクト・ネットワーク上の全部または一部を特定して自己の意図を指示する。

【0044】(S5)：この状況をコミュニケーション・マネージャ10が識別し、カーネル12を介在させて、サーバに対して処理対象となる位置に該当するオブジェクトについてのテンプレートを呼出させる(サーバがテンプレートを呼出す)。

【0045】(S6)：サービス・モジュールは当該テンプレートに対応した領域準備を行う。

(S7)：コモン・プラットフォームにおいては、当該テンプレートに記述されていることのある制約データを抽出して、当該制約データをデータ・ウインドウ(II)9-2に表示する。

【0046】(S8)：クライアント1は当該制約データを見て、その中から所望する制約データを選択する。

(S9)：この状況をコミュニケーション・マネージャ10が識別し、カーネル12を介在させ、サーバに通知し、サーバが実行計画を実行する。

【0047】(S10)：これにともなって、サービス・モジュールが所望される処理(例えば、線引きや色付けなどの処理)を実行する。

(S11)：処理結果をデータ・ウインドウ(I)9-1に表示する。

【0048】(S12)：クライアントはその処理結果を評価し、次の指示を行う。図7はコミュニケーション・マネージャの働きを示す図である。図中の符号は図1に対応している。なお図7においてウインドウ4に対応して「コモン・プラットフォーム」というラベルを付しているが、図1に示すコモン・プラットフォーム2においてクライアント1から直接見える所がウインドウ4であってコモン・プラットフォーム2を代表しているとも考えられることから「コモン・プラットフォーム」とラベルを付しているだけであり、図1に対応するとウインドウ4に該当しているものである。

【0049】図7に示す如くクライアントの意図がウインドウ4を介してシステムに伝えられるときには、必ずコミュニケーション・マネージャ10を経由してカーネ

ル12や各マネージャ14, 15, 16更にはサービス・モジュール17-iに伝えられる。このことから、コミュニケーション・マネージャ10はウインドウ4に直結している唯一のマネージャとなり、対話プロセスを集中的に管理している。当該コミュニケーション・マネージャ10の働きについては更に詳しく後述される。

【0050】上述の如くWELLは分野を問わない拡張性のある言語である。それぞれの分野に対応するウインドウ定義やサービス・モジュール定義といったオブジェクト・ネットワークを与えると、WELLはその分野に対応する形で分野記述言語となる。このWELLを用いたウエル・システムは、ウエル・カーネル12と各種マネージャ14, 15, 16など(ウエル・マネージャ)との2つの要素から構成される。

【0051】エキステンシブ・ウエル・カーネル12は、以下の項目を行う。

(1) データ状態の把握(プロセスの把握を行い、次に実行すべき関数の選択)

(2) イベントの解析(各ウインドウから送られてくるイベントを判別し、その結果に対応するシステム内のプロシーダをコール)

(3) 関数実行要求(総称的関数に制約条件を付加し、関数の実行をリクエスト)

また各マネージャは以下の4つに分けられる。

(1) コミュニケーション・マネージャ10(テンプレート(データ)の間接的管理及びコモン・プラットフォームからの入力に対する各マネージャの制御)

(2) 関数実行マネージャ16(関数の実行管理)

(3) ウインドウ・マネージャ14(ウインドウの管理)

(4) ディスプレイ・マネージャ15(テンプレートの数値変換)

分野に依存しないWELLと分野依存のサービス・モジュール17-iとのデータ変換は、コミュニケーション・マネージャ10が行う。このコミュニケーション・マネージャ10を介することによって、あらゆる分野のデータを扱うことが可能となり、ウエル・システム自体の汎用性は失われないこととなる。

【0052】ウインドウ4はクライアント1との間のインターフェースを行うものであり、以下の5つの個々のウインドウが存在する。

(1) ネットワーク・オペレーション・ウインドウ5(システム全体の制御)

(2) オペレーション・ウインドウ6(ネットワークのオペレーション表示及びその制御)

(3) コマンド・ウインドウ7(選択候補の表示及びキーボードからの入力)

(4) メッセージ・ウインドウ8(システム側からのメッセージ表示)

(5) データ・ウインド9-1, 9-2(データの表

10

20

30

40

50

示)

言うまでもなく、ある役割をプロセッサに導入する時、オブジェクト・ネットワーク中の個々のオブジェクトについての当該オブジェクトの表現方法は、役割依存でなければならない。人間に対しては、可視オブジェクトであり、計算機にとってはデータスキーマに則したデータ構造によって、オブジェクトが表現される必要がある。オブジェクト・ネットワークは個々のプロセッサで共有するが、データ構造は、テンプレート形式で定義されて、相互変換可能とされる。

【0053】オブジェクト・ネットワークを構成する各オブジェクトとしては、分野記述言語の枠組みに従って、名詞オブジェクトと動詞オブジェクトと制約オブジェクトとされ、制約オブジェクトは、被制約オブジェクトと制約条件とが属性として与えられ、そのための妥当性が各マネージャによって判定されて対応するオブジェクト処理を行いうるようになされる。

【0054】上述のウエル・システム11にカラー画像生成・色付け処理分野をあてはめると上記WELLはいわばWELL-PPP (Window-based Elaboration Language for Picture Processing and Painting) となる。そしてウエル・システム11はカラー画像生成・色付け処理システムとなる。この場合の1つのオブジェクト・ネットワークである要素ネットワークは、個別オブジェクトの外観を表しかつ領域を確保している輪郭 (Outline) を生成する輪郭部 (Frame Section) と、ラインとそれによって確保されている領域にしたがって属性値 (Attribute value) を与えるカラー部 (Color Section) とからなる。

【0055】図8は要素ネットワークの関係を示す図である。図8における左半分の「フレーム」20は上述の輪郭部 (Frame Section) に該当し、右半分の「カラー」21は上述のカラー部 (Color Section) に該当している。

【0056】輪郭部20ではデータ自身を表す項目 (Entity) を作成し、カラー部21では輝度、色度などを表す属性 (Attribute) を作成する。各クラスの状態は以下のようになる。

<NONE>まだ処理の行われていない状態を表す

<POINT>輪郭の基本となる点を示し、これを主要点 (Principal Point) と呼ぶ

<POINT SEQUENCE>主要点に対し、その接続順序を示す

<LINE SEGMENT>接続順序に基づいてスプライン関数で補間処理がされた状態

<LINE>LINE SEGMENTを接続して1つのラインを完成した状態

<REGION SEGMENT>画像におけるその水平走査線1本分の領域を示す

<REGION>REGION SEGMENTを画像

に沿って上下方向に接続したものを示す

<COLORED>輝度、色度情報を示したそれぞれの項目を表す。また輪郭部とカラー部を結ぶIDENTIFYは以下のように定義される。

【IDENTIFY】処理の対象を特定する作業を示す。

【0057】図8に示す如きネットワークを利用してクライアントが絵を描く処理を行うことができる。当該処理の態様に関しては、本願発明と直接関連するものではないが、先に特許出願している特願平4-34864号 (オブジェクト・ネットワークによる言語処理システム) にも一部記述されている。

【0058】図9はコモン・プラットフォームをもつシステムにおいてカラー画像生成・色付け処理を行う態様を示している。図中の符号1, 2, 3は図1に対応している。

【0059】コモン・プラットフォームにおけるウインドウの表示によって、視覚的に要素ネットワークが組み込まれたことがクライアントにわかると、クライアントは描画するために必要な意図を選択する (クラス選択)。この例では点に輝度・色度を与えるCOLORED POINTを選択する。

【0060】点に色をのせたいというクライアントの要求に対して、サーバは情報を格納するためのテンプレートを準備し (DEFINING PROCESS)、どの点に対して色情報をのせたいのかクライアントに要求する。テンプレートについては後述される。

【0061】クライアントが点を指定すると (IDENTIFY)、サーバはインデックスを参照することでその点を認識する。そしてその点にのせる輝度・色度データをクライアントに要求する。

【0062】クライアントからデータを応答として受け取るとDEFINE OPERATIONとしてテンプレートにデータを入力し、その結果としてのカラー画像をウインドウに表示する。

【0063】この表示によってクライアントは、そのプロセスを認識し、それを評価し、次の意図選択へ移ることができる。このようにクライアントはサーバからの意図を可視化オブジェクトとして把握する。

【0064】上述の如く描かれる点や線は夫々上述の名詞オブジェクトの形で管理され、また個々の名詞オブジェクトに対して行われる処理態様は上述の動詞オブジェクトとして与えられる。

【0065】上述のコミュニケーション・マネージャ10は、WELLの拡張性を維持するためにデータ相互関係を管理する。そのためにデータは、テンプレートとして一元的に管理され、そこに属性を自由に設定できるようにされている。

【0066】図10はテンプレートのフォーマットの一例を示す。個々のテンプレート22がその属するオブ

エクトについての番号に対応するかを示すインデックスと、座標と、属性とをもつ。テンプレートは、サービス・モジュール17-i内では「インデックス」によって管理されるが、データ・ウィンドウ9を通してクライアントから与えられるものは、データ・ウィンドウ9上の座標値である。サービス・モジュール17-iは、座標値からテンプレートのインデックス検索を行えないので、コミュニケーション・マネージャ10は座標値をインデックスに変換し、それをサービス・モジュール17-iに引き渡す。更に、コミュニケーション・マネージャ10は、ウエル・システムの持つ対話プロセスを集中管理する。コモン・プラットフォームは、(i)クライアントにオペレーションを伝えるオブジェクト・ネットワーク、(ii)メッセージや選択項目を伝えるメッセージ・ウィンドウ、(iii)実際に描画などの作業を行い、画像データなどを表示するデータ・ウィンドウなどを持つが、それらのウィンドウを通して得られたクライアントの意図を実現するために、コミュニケーション・マネージャ10は、その意図内容の入力に応じて、それを各マネージャやカーネルへ伝える。コミュニケーション・マネージャ10のこのような対話プロセスの一元管理により、データが必ずここを通過することになり、テンプレートのチェックを行ったり、制約条件のチェックを行うことも可能になってくる。

【0067】図11はコミュニケーション・マネージャの動作態様を示す。図中の符号1, 3, 5, 8, 9, 10, 12, 14, 15, 17は夫々図1中の符号に対応している。

【0068】ウィンドウ上でのクライアントによる指示点(x, y)などがデータ・ウィンドウ9からコミュニケーション・マネージャ10に伝えられ、メッセージはメッセージ・ウィンドウ8を介して伝えられ、オペレーションはオペレーション・ウィンドウ5を介して伝えられる。そしてそれらは夫々コミュニケーション・マネージャ10によって各種他のマネージャ14, 15やカーネル12やサービス・モジュール17との間で情報の送受が行われる。

【0069】図10に示す如きテンプレートは、点や線のデータ、図形/画像データなどに対応して用意されるものである。これは、点や線のデータ、図形/画像データなども、コモン・プラットフォームにおいて取り扱われるデータであることによる。それらは、オブジェクト・ネットワークの中でノードを形づくる名詞オブジェクトに属している。

【0070】したがって、データをクライアントが表示させあるいは参照しようとする場合、ウィンドウ4上に表示されているデータの参照部分をマウスなどで示すと共に、名詞オブジェクトをオブジェクト・ネットワーク上で参照指示する。

【0071】このような参照指示機能は、クライアント

が動詞オブジェクトの目的格としてのデータを指示することを主眼とするものである。これに対して、実体表示に当たっては、サーバが実行処理した結果をクライアントにみせるのに実際の点、線、図形、画像を実体としてウィンドウ4上に見せるようにする。サーバ側ではデータを蓄積し、処理を行うに当たって、具体的なデータ構造(例えば、インデックス付加リスト構造、ラスタ構造など)に沿って蓄積したデータに対して処理を行う。

【0072】図12はクライアントの指示に対応する動作を説明する図である。図中の符号1, 4, 10, 12, 15, 17-iは夫々図1に対応し、23はカーネル12内にもうけられている作業用のバッファ、23'はバッファ23上の内容例、19Cはオブジェクト・ネットワークを表している。

【0073】今、カラー画像生成・色付け処理がある程度進行しているとする。この場合、画像生成・色付け処理に対応するオブジェクト・ネットワーク19Cから、該当するサービス・モジュール17-iに対して、点や線などに対応するテンプレートがデータとして与えられ、当該サービス・モジュール17-iからカーネル12上(なお、バッファ23はコミュニケーション・マネージャ10上に位置していてもよい)のバッファ23に対して、図示のバッファ内容23'の如く、テンプレートのコピーが書き込まれている。即ち、例えば(i)インデックス「1」で特定される点の座標が(50, 40)であって色度や輝度(属性で示される)については未定(?)で表している)であり、(ii)インデックス「2」で特定される点の座標が(80, 250)であって色度や輝度は未定であり、(iii)インデックス「3」で特定される点の座標が(270, 400)であって色度や輝度は未定であり、...の如き、各テンプレートのコピーがバッファ23上に記述されている。

【0074】この状態の下で、いま、クライアント1がウィンドウ4上で例えば座標(80, 250)を指示して1つの点を特定したとすると、コミュニケーション・マネージャはその動作を検出し、座標値についてウィンドウ・マネージャ14が検出してコミュニケーション・マネージャ10に通知する。したがって、コミュニケーション・マネージャ10は当該指示された点の座標値(80, 250)と対応するインデックス値を検索する。すなわちコミュニケーション・マネージャ10はバッファ23の内容にもとづいて座標(80, 250)をもつテンプレートに対応するものがインデックス「2」であることをコミュニケーション・マネージャ10は知り、その後の処理に移る。

【0075】したがって、輪郭部で打たれた主要点の全座標を作業領域にコピーしたテンプレートと送られた座標との照合が行われる。一致もしくは近似の座標を検索するとそのインデックスを特定し、コミュニケーション・マネージャ10はサービス・モジュール17-iに送

り制御を行う。サービス・モジュール17-iはそのインデックスをもとにそのテンプレートの値を更新する。例えば座標位置を変更したり、属性上での輝度値を記述したりする。当該更新されたテンプレートはディスプレイ・マネージャ15によって可視化オブジェクトに変換されウインドウ4に表示される。この場合は打たれた点がクライアントに認識できるように反転し浮き上がる。このようにエキステンシブ・ウエルは常にインデックスを中心に管理しているサービス・モジュール17-iと情報交換をする。こうすることで特定の分野に依存しない拡張性の高いシステムが実現できるとともに、どのような分野に対してもクライアントはサーバの意図をディスプレイ・マネージャ15によって変換された可視化オブジェクトをみて操作できる環境を提供する。

【0076】図13は情報の交換動作を説明する図である。図中の符号1, 4, 6, 9, 10, 12, 14, 16, 17は夫々図1に対応している。クライアントがサーバと情報のやりとりを行う際、クライアントは常にウインドウを介して情報のやりとりを行う。そのためシステムは常にウインドウで待ち状態になっていなければならない。クライアントの意図によって駆動される駆動(Event Drivenという)にたいして、システムの制御をウインドウでの待ち状態にする駆動をData Drivenという。Colored Pointのクラスが選択されていてそれを駆動するのがEvent Drivenである。しかし輝度・色度のデータが与えられなければ、関数は実行できないのでウインドウにてクライアントの入力待ちとなる必要がある。この駆動がData Drivenである。

【0077】図13において、クライアント1が或る座標上の点を指示して上記イベント・ドリブン(Event Driven)を行ったとする。この状況の下で、「コミュニケーション・マネージャ10によってウインドウ・マネージャ14に通知された上でウインドウ・マネージャ14が座標値(例えば80, 250)を報告し、対応するインデックス「2」が調べられてサービス・モジュール17-iに通知される」ことは、図12に関連して説明した通りである。

【0078】今、クライアント1が当該点に対して色付けを行おうとしているとすると、図8に示したクラス「Colored Point」の状態分析をカーネル12が行う。

当該分析を行ったカーネル12は関数実行マネージャ16に対して関数実行要求を発する。これによって関数実行マネージャ16は与えられる制約条件(Constraint)によって具体的関数を特定する。サービス・モジュール17-iは特定された具体的関数にもとづいて処理が実行可能であるかを判別し、当該処理に必要なデータが不足していればカーネル12を介して必要な作業を行う。例えば今、サービス・モジュールはクライアントに特定された点のテンプレートに輝度・色度情報を入力するようにしているとすると、輝度・色度データが未定である

ため、関数の実行ができない。ここでData Drivenが駆動され、この場合データ入力に必要なChrominance DiagramとLuminance Diagramが開かれ、輝度見本や色見本が表示され、クライアントの入力待ちとなる。

【0079】サービス・モジュール17-iは必要なウインドウ(図13に示すDIAGRAMのためのウインドウ)を開いて、ユーザの応答を待つが、その応答以後そのウインドウが不要になる場合がある。またサービス・モジュールによっては多数のウインドウを必要とするものも考えられるため、表示されるウインドウは必要最小限にするべきである。そこでウインドウ・マネージャ14内にウインドウの状態を示すテーブルを用意し、不要ウインドウの自動消去を行う。

【0080】図14はチェック・モジュールの動作を説明する図である。図中の符号8はメッセージ・ウインドウ、9はデータ・ウインドウ、22はテンプレート、24はチェック・モジュール、25は制約条件、26はオブジェクト・ネットワークを表している。

【0081】クライアントは、オブジェクト・ネットワークに従った作業の流れの中で、自分がデータ・ウインドウ上に生成していったデータの正当性を確認したいと思うことがある。そのようなクライアントが確認したいと思う事柄の一つの制約条件として捉える。このような制約条件25は、オブジェクト・ネットワーク26のそれぞれの名詞オブジェクト(テンプレート22で表されるデータの状態)に付属的なものとして課され、サーバ側では、その制約条件とテンプレートの内容とを比較することで確認作業を行う。ここで問題にされるのは、サーバが、処理対象とされるテンプレートの特長方法(Identify)をどのように持つべきかである。ウエル・システムでは、コミュニケーション・マネージャの機能を利用してこれを実現できる。コミュニケーション・マネージャ10は、オブジェクト・ネットワークの持つテンプレートの内容データによってクライアントとサーバとの間のデータの整合性を管理しているので、クライアントがマウスでヒットした座標値(X, Y)から上記図12を参照して説明した如く処理対象を特定するための情報をテンプレート22から引き出して、チェック・モジュール24(具体的に制約条件をテンプレートに照らし合わせるサービス・モジュール)に渡すことが出来る。チェック・モジュールでは、その情報を頼りに、その制約条件が満足されているかどうかを確認し、その結果をメッセージ・ウインドウを通してクライアントに知らせる。このようなチェック・モジュール24によるチェック動作は、図8において例えばクラス<POINT>の状態からクラス<COLOREDPOINT>の状態へ移行した際に、正しい形で移行が行われたか否かをチェックするためなどに利用される。更に他の制約動作も、他のサービス・モジュールと同様に対応するマネージャを用意することで実現される。

【0082】図15は制約条件の表示態様を示す図である。図中の符号5, 8, 9, 10は図1に対応し、25は図14に示される制約条件、27は名詞オブジェクトを表している。

【0083】図形描画の間に、例えば描かれたライン(線分)に対してどのような制約条件が付されているかをクライアントに知らせる必要があることがある。このような場合の各マネージャの動作が図15によって説明される。

【0084】上記ライン(線分)は、それ自体がデータの状態を表していることから、名詞オブジェクト27の1つとして捉えることができる。上記制約条件25は名詞オブジェクト27に対応して付されているが、ライン(線分)という状態に対して課され得る制約条件25として考えられるのは「連続性(Continuity)」と「対称性(Symmetry)・・・(註)当該線分がある軸または点に対して対称な状態にあるか否か」といったものである。

【0085】名詞オブジェクト27がもつこのような制約条件25はコミュニケーション・マネージャ10の介在の下でメッセージ・ウインドウ8に表示されてクライアントに知らされる。なお図15において、メッセージ・ウインドウ8の内容に関して「Symmetry-Point, Axis」とあるのは「点対称」か「線対称」かを問う表示である。当該表示に対応して、クライアントは「Point」か「Axis」かをピック・アップするようにして意図をサーバ側に通知する。

【0086】図14に示すチェック・モジュール24は、メッセージ・ウインドウ8を通しての起動要求をクライアントから受けるとファンクション・マネージャによって起動される。チェック・モジュール24は、必要があればクライアントに対してアイデンティファイ作業の要求をメッセージ・ウインドウ8を用いて行う。クライアントは、それに対してマウスでデータ・ウインドウ上の可視化対象をヒットすることでコミュニケーション・マネージャ10へその対象についてのデータ値の引き渡しを行う。このデータ値は、コミュニケーション・マネージャ10によって図12に示す如く対象に相当する項目データのインデックスに変換されるとチェック・モジュール24へ送られ、このインデックスを使ってチェック・モジュール24は、テンプレートから具体的な属性値を引き出してチェックを行う。チェックされた結果は、メッセージ・ウインドウによってクライアントに知らされる。

【0087】図16はアイデンティファイ作業を説明する図である。図中の符号4, 5, 8, 9, 10, 14, 17-iは夫々図1に対応している。上述の如くチェック・モジュール24がクライアントに対してアイデンティファイ作業の要求を行うと、クライアントはデータ・ウインドウ9上で、図16に示す如く、例えばあるライ

ン(線分)をヒットする。これに対応して図12に示した如くウインドウ・マネージャ14の働きによって関連しているデータがコミュニケーション・マネージャ10に知らされ、コミュニケーション・マネージャ10が、対応するインデックスを知り、当該インデックスをチェック・モジュール24に通知する。これによって、図14に示した如く、チェック・モジュール24がチェックし、その結果がメッセージ・ウインドウ8において表示される。

【0088】なお、図形描画における点や線が名詞オブジェクトとしてデータの1つとして把握され、図9に示す「単位意図(要求)」などの要求が動詞オブジェクトとして対応づけられる。そしてそれらの名詞オブジェクトや動詞オブジェクトが夫々テンプレートの形で表現される。

【0089】上述した構成と動作とについて、より具体的に例を挙げつつ説明を補足する。図2において、クライアント1の意向に従ってサービス計画を準備するようにする。例えば、図17に示す如き魚のうろこパターンのようなテキストチャを描画したいとの意図をクライアントが持つとき、テキストチャ描画の意図をサービス依頼(要求)としてエイジェント・ロール・サーバ3'に伝え、エイジェント・ロール・サーバ3'は流れ線(flow lines)とセル画像(cell picture)とをクライアントと協力して描画するサービス計画を具体的に準備する。流れ線を書くには、図8の要素ネットワークがある要素画像を描くために利用できる、それを利用する。すなわち、図8中でPOINTからLINEまでの部分ネットワークをまずオブジェクトネットワークとして表示する。

【0090】それは図2に示す役割サーバ検索(事項検索)が流れ線(flow lines)のlineの事項を検索すると、相当する単語が要素ネットワーク中に存在することを検索し、その部分オブジェクト・ネットワークを、COMMONプラットフォーム2に「flow line」の表示と共に表示する。

【0091】クライアント1は当該表示に従って図17(a)に示すように魚体を示す点線を沿う形で曲線AとBとを描く。次に図17(b)に示すように、線C, Dを描く。このとき、How many points do you divide on C and D?の如き質問がエイジェントからメッセージ・ウインドウ8に表示され、それに対してクライアント1が値「4」とキーインして答えると、図17(c)に示すように、4個の補間曲線が引かれる。次のステップで質問How many points do you divide on A and B?に対し値「16」と答えると、各補間曲線上に図17

(d)に示す如くうろこを描くための16個の中心点が表示される。この場合、Flow linesについての制約条件として隣接するFlow linesとの間の距離に比例して中心点間距離が大きくなるような制約条件が与えられるとすると、即ち、Flow lines間隔が狭ければ、Cell Picture

としての「うろこ」の寸法が小さくなければならないとすると、このため次のSatisfying文制約記述が与えられる。

Satisfying :Every length between adjacent points a long flow line is proportional having a fixed ratio with vertical length at the point between adjacent flow lines,end.

それと同時に「cell picture」の表示と共にうろこパターンを画像を描くようにFlow linesと同時に要素ネットワークとしてのオブジェクト・ネットワークが表示され、クライアントはそれに従って図18に示す如く、上記中心点(×印)を中心としてうろこのカラー画像を描く。

【0092】この処理は、図5の処理フロー(I)に従って実行される。この実行をある特定のスペシフィック・ロール・サーバ3"に割り当てるために、図2に示すサービス役割割り当て計画(サービス仕様)によって実施すべき処理内容があるサーバに要求される。

【0093】それが図2に示すサービス役割機能(要求)として計画されたスペシフィック・ロール・サーバに実行要求が発行される。この発行はエイジェント・ロール・サーバ3'とスペシフィック・ロール・サーバ3"との間での共通の場を通じて機能する。

【0094】エイジェント・ロール・サーバ3'に依存した図2に示す従属表示のプロセスで所定の表示が行われて、エイジェント・ロール・サーバ3'とスペシフィック・ロール・サーバ3"との交信が行われ、当該従属表示のプロセスがコモン・プラットフォームとして、エイジェント・ロール・サーバ3'とスペシフィック・ロール・サーバ3"群との間の連絡通信の場を提供する。したがって、エイジェント・ロール・サーバ3'が複数のスペシフィック・ロール・サーバ3"に依頼した複数の処理内容、この例では「Flow lines」と「cell picture」との間に存在する整合性についての制約条件について規定し、実行する必要がある。

【0095】そのための前提条件として、予めテンプレートについての定義を行うことを必要とする。点についてのテンプレートは図10のように定義され、そこでの属性値集合としてのattributes for point(s)(x,y)として、当該点に関する輝度、色度などの属性値を入れるべきデータ形式が示されている。Lineについては、走査線インデックスを"i"とすると、"i"についてのリスト構造として(x,y_i)の値が、図19に示す如きLine用テンプレートが定義される。

【0096】即ち、当該Lineが走査線と交差する各点のうちの主要点No.1, No.2...No.nについてのテンプレート(図10参照)上のattributes for point(s)(x,y)に、入口側のポイントpと当該点の輝度と当該点の色度ベクトルと出口側のポイントpとが記述され、上記夫々のポイントによって他を指示することによって、全体が1つ

のLineに対応するLine用テンプレートとして定義される。なお、図19に示す各点の属性値に輝度と色度ベクトルとを記述しているのは次の理由による。即ち、PointとColored pointとがオブジェクト・ネットワーク中に定義されているような要素ネットワークでは、最初からPointのテンプレート中に、Coloredの修飾形容詞に対応した輝度値や色度ベクトル値を属性値として持たせることが望まれるからである。

【0097】FlowというLineを限定する形容詞節が付加されると、そのためにテンプレートに付加属性のためのスペースを必要とする。このスペースの設定は予め行われてもよいし、エクステンシブルWELLの機能の拡張機構としてのグラフ構造エディタを用いて必要時点で設定してもよい。

【0098】図17(c)に示した流れ線を示すようにする当該形容詞節で特定するデータ構成としては、図20に示す如きものが用意される。図20における符号30はFlowという「Lineを限定する形容詞節」につけられたインデックスであり、符号31は当該Flowに属するLineについてのインデックスであり、符号32は当該Lineについてのデータ配列(図19に示す如きもの)である。

【0099】Colored pointについてFeatureという形容詞をつけて、それをFeatured colored pointとすると、テンプレートの属性の所に、図21に示す如く、Feature名を付加する。Feature名の所には、"edge", "highlight"または"none"を記述するようにする。"edge"が記述されると、その点が輝度や色度ベクトルからエッジと識別されたことを示す。"highlight"が記述されると、その点がハイライトとして示されることを示す。また"none"が記述されると、その点にはFeatureを持たないことが示される。

【0100】そのいずれにしても、Flow linesに対して図17に示したようなA, B, CおよびDの4個のLinesに対応して、夫々のLine用テンプレートが準備される。次にクライアント1とコモン・プラットフォーム2中のメッセージ・ウインドウ8との通信によって更に4本のFlow linesのためのLine用テンプレートが設定される。次に"cell picture"を描画するために、オブジェクト・ネットワーク(図8に示す如き)に対応したオブジェクトに対応してテンプレートが設定され、最後のColored Regionでは輝度色度データをもつラスタ形式のテンプレートを持つようになる。

【0101】このように設定された「うろこ」画像を、流れ線上に設定された中心点に「うろこ」の中心点を合わせ、サイズを選んで写像する。この際、制約条件を必要とする。

【0102】この時点までにおいて、Interval between the adjacent center points on the flow linesが各center pointに定義されている。即ち、各flow lineが

もつ属性としてIntervalが定義されており、flow line のdrawing の実行後計算され、属性値として、各center points についてのテンプレート中にそのInterval値が格納されている。

【0103】そして同様に、cell pictureのもつ属性としてscale をもち、そのscale 値とinterval値との間の制約関係として、

a scale of cell picture = the interval between the adjacent center points on the flow line

を設定する。そしてこれを満足するように、mapping を行うスペシフィック・ロール・サーバ3" にそのmapping を依頼する。

【0104】図22と図23とは上下に組合されて1つの流れを表すテキストチャ描画の流れ図を示している。クライアント1の意図する所が、図22、図23における左端に、上から下に向かう時系列で示される。またエイジェント・ロール・サーバ3' の処理が、図22、図23の左右中央部に、上から下に向かう時系列で示される。さらにスペシフィック・ロール・サーバ3" の処理が、図22、図23における右端に、上から下に向かう時系列で示される。また図22、図23における符号1, 2-1, 2-2, 3', 3" は図2に対応している。更に図22の下端における矢印(A)は図23に示す矢印(A)につらなるものである。

【0105】クライアント1がテキストチャ描画を意図してテキストチャ・サービスを要求すると、エイジェント・ロール・サーバ3' はテキストチャ描画サービス計画を準備し、統合サービス・プラットフォームを設定する。これにもとづいて、クライアントは図17に示した如きFlow line を表示させようとして、Flow line 表示を要求する。

【0106】エイジェント・ロール・サーバ3' はフローライン計画を検索し、スペシフィック・ロール・サーバ3" に対してフローライン実行サービス設定を要求する。スペシフィック・ロール・サーバ3" の設定した結果が、エイジェント・ロール・サーバ3' にて中継されて、クライアント1に示される。

【0107】クライアント1がFlow line 描画を意図して、フローライン用画面を要求すると、エイジェント・ロール・サーバ3' からスペシフィック・ロール・サーバ3" に依頼されて、クライアント1に対してフローライン画面が表示されると共に制約条件の提示を要求する。

【0108】制約条件について応答が了承されると、エイジェント・ロール・サーバ3' はフローラインとセンタ点とを表示する計算を行って、クライアント1に対して表示する。クライアント1は必要に応じて修正要求を行う。

【0109】次いでクライアント1はCell pictureについての描画を意図し、図23に示す如く、エイジェント

・ロール・サーバ3' とスペシフィック・ロール・サーバ3" との処理によって、セル・ピクチャ用のコモン・プラットフォームが設定される。

【0110】クライアント1がセル・ピクチャの描画を意図すると、同様に、セル・ピクチャの画面が表示されかつ制約条件の提示が要求される。クライアント1が制約条件について応答すると、セル・ピクチャについての処理結果が統合サービス・プラットフォームに統合され、クライアント1の評価を待つようにされる。

【0111】ここでは、フローライン描画とセルピクチャ描画との二つの仕事は並列に実行可能になっている。並列実行可能な動詞オブジェクトを検出するためのシステム構成は上記テキストチャ描画の例では、テキストチャを描くためのフローラインのための線を描くことと、フローライン上にセル画像を描くこととを並列に行い得るようにされる。そしてその結果がそろった所で、統合描画プロセスに移る。従ってスペシフィック・ロール・サーバ3" としてはフローライン用、セルピクチャ用、統合描画用の3個のサーバがエイジェント・ロール・サーバ3' に付属している。これらを制御するオブジェクト・ネットワークとしては図8の要素ネットワークを使用し、フローラインとしては図中右側のNoneからLineまでを使用する。これはFlow line のlineを描くことを目的としていることから、この部分のみがコモン・プラットフォームに図示され、表示のプロセス2-1中でクライアント1が使用する。Cell pictureと統合描画との両部はpicture を描画することから、図8全体がオブジェクト・ネットワークとしてクライアントに使用可能である。

【0112】これら個々のスペシフィック・ロール・サーバ3" における実行結果は、エイジェント・ロール・サーバ3' を通じて、クライアント1とエイジェント・ロール・サーバ3' との間に存在するコモンプラットフォーム上にその実行結果としての画面表示される。

【0113】得られた画像表示をクライアント1は観察しながら、エイジェント・ロール・サーバ3' はその仕事に関して必要とする属性もしくは制約についてクライアントに質問を行う。このための構成が図14に示されている。図14に示す如く実行され確認され、data identify の後に、コモン・プラットフォームのデータ・ウィンドウ9に表示され、クライアントから応答された制約がチェックされ、その実行がValidity check 24によっておこなわれ、メッセージ・ウィンドウ8でその実行が確認されて、結果がデータ・ウィンドウ9に表示される。

【0114】これらの特定サービスは、エイジェント・ロール・サーバ3' からスペシフィック・ロール・サーバ3" へ、その間に存在するコモン・プラットフォームを用いて依頼される。このコモン・プラットフォームが図22、図23においては従属表示のプロセス2-2と

して示され、エイジェント・ロール・サーバ3'に従属しているものとして示されている。この場合の態様は、ソフトウェアの仕様からオブジェクト・ネットワーク・データをはじめとするすべては、クライアント1とエイジェント・ロール・サーバ3'との間の場合と全く同じ形式をとる。それは、システム開発の際のユーザフレンドリ性を確保するためである。

【0115】インタラクションの形態としては、

(a) エイジェント・ロール・サーバ3'のような抽象化サービスはジェネリック・オブジェクト・ネットワークとして抽象化されている。当該ジェネリック・オブジェクト・ネットワークは、パラメータあるいは制約条件によって具体化されたスペシフィック・オブジェクト・ネットワークと階層化される。なおスペシフィック・オブジェクト・ネットワークは特定の機能に対応したサービスを行うネットワークである。

(b) 個々のオブジェクト間の属性は、親の方から（例えばエイジェント・ロール・サーバ3'）から受け取り、それを用いてスペシフィック・ロール・サーバ3''は実行し、その結果は合成された属性(synthesized attribute)として、依頼した親の方に受け渡され、エイジェント・ロール・サーバ3'はそれらの属性値を用いて、制約条件を付し、新しいデータを得たり、チェック機構を動かす。

(c) 制約条件などは、図4に示す如く、形態的制約と時相的形態とが存在する。

【0116】ジェネリック・オブジェクト・ネットワークは、図8に示す要素ネットワークの如きものでテキスト画像を図17、図18のように描くため、クライアントがテキスト描画の意図からその要求をエイジェント・ロール・サーバ3'に与えると、エイジェント・ロール・サーバ3'のもつジェネリック・オブジェクト・ネットワークを、統合サービスプラットフォームとして、クライアント1とエイジェント・ロール・サーバ3'との間の意志疎通のために設定する。そしてその上に上記図17や図18を表示するようにする。このようなネットワークは、エクステンシブル WELL のカーネル部の機能としてのグラフ構造エディタで行われる。

【0117】図24はオブジェクト・ネットワークの階層構造を示すものである。例えば数学において変数を一般化して与えて公式を用意しておくように、パラメータや制約条件などを一般化して（抽象化して）与えておいたジェネリック・オブジェクト・ネットワーク33を用意しておく。そして、特定の処理を行わせるための特定のパラメータや制約条件34を、ジェネリック・オブジェクト・ネットワーク33中に組み込むことによって、当該特定の処理のためのスペシフィック・オブジェクト・ネットワーク35がつくり上げられる。

【0118】図25はエイジェント・ロール・サーバ3'からスペシフィック・ロール・サーバ3''への処理

の詳細化をテキスト描画の場合についてのジェネリック・オブジェクト・ネットワークを例示した図である。

【0119】そして図25中で四角形で示した名詞オブジェクトと関連したテンプレートがテキスト描画システムの設計を行ったエキスパートによって準備され、それをWELLの機能によってエイジェント・ロール・サーバ3'もジェネリック・オブジェクト・ネットワークとデータとに分けてもつようにされる。

【0120】Flow line の名詞オブジェクトがクライアントにより指示されると、図8の要素ネットワークのうちLINE~NONEの部分グラフがFlow line のための部分描画プロセスとして、フローライン用コモン・プラットフォームがスペシフィック・ロール・サーバ3''からエイジェント・ロール・サーバ3'を中継して、クライアント1に指示され、これを用いて図17(a)(b)

(c)(d)のように描画プロセスが進行するようにされる。

【0121】この際、テンプレートは、要素ネットワークのもつテンプレートがそのまま利用される。FlowというLINEについた修飾形容詞は、LINEに新しい属性構造をテンプレートとしてもち、さらに、(i) 制約条件を付加すべきこと、および(ii)テキスト描画のためにはLINEが描画用データウインドに表示されるのではなくセルピクチャを写像する図26に示すような点の位置のみが必要であることを、を示している。

【0122】以上のことから、Flow line を描画するためには、図17(b)をもとに図17(d)を描き、さらに図26を描画するための制約条件、もしくはサブネットワークを与えるようにされる。

【0123】したがってFlowという形容詞による修飾内容についての枠組みとして、前述したように、与えられた制約条件のもとに、

Length between adjacent points along flow line

Vertical Length between adjacent flow lines

について、エイジェント・ロール・サーバ3'はクライアントから定義してもらう。このために、メッセージ・ウインドウ8によって前述したような分割する点の数についての質問応答が行われる。その結果が図14のConstraints on nounobject 25に与えられ、Flowの修飾が行われ、図26の結果が得られる。図中の点はCell pictureを写像する位置を知らせるためのもので、実際のTexture 画面には表示されない。これはTexture とFlow line の両者が与えられることから、Texture picture 表示の際、消去される。

【0124】Cell pictureについても同様で、pictureによって要素ネットワークによってカラー画像が描画される。この描画の後、図26の点集合中の各点の周りに写像されるが、picture はCellによって修飾されており、その大きさをFlow line での隣接点間距離との関係によって規定する。そのためにCell pictureに付属する

制約としてCell pictureの2つのlengths がFlow line について定義されているlengthと一致させる処理が行われて写像(mapping) 描画がTexture picture によって行われる。この際、Cell picture size とFlow line のlengthとの比を設定することが可能である。これは図4に示される制約意図の実現と考えられ、サービス間形態調整制御が実際に整合化を行う。

【0125】以上のことを実行するために、Texture picture を描画対策としての要求は、Request to paint texture pictureがクライアントから意図として、図2 2、図23に示されているように、エイジェント・ロール・サーバ3' に与えられると、エイジェント・ロール・サーバ3' は、統合サービス・プラットフォームをサービス計画準備のプロセスとして設定し、その上に図25で示されるジェネリック・オブジェクト・ネットワークをデータベース検索によって取り出し、それと同時にflow line とcell pictureとのためのスペシフィック・ロール・サーバが割り当てられる。

【0126】以上のスペシフィック・ロール・サーバのなすべき仕事については、図2に示されるサービス計画準備で、役割サーバ検索(事項要求)が、エイジェント・ロール・サーバ3' を制御しているWELLシステムのカーネル部を介してデータベースサーバに検索依頼し、割り当てられたスペシフィック・ロール・サーバにその仕事を割り付ける。あるいは、元来その仕事を行っていたスペシフィック・ロール・サーバを事項検索を行って見出し、それに依頼する。

【0127】スペシフィック・ロール・サーバが行う役割は基本的な名詞オブジェクトの実現を行うサービス実行と考えられ、その役割は具体的オブジェクトネットワークによって定義される。これに対しエイジェント・ロール・サーバは、それに制約条件や、他との整合化を行う仕事で、クライアントとの間でのインタラクティブな対話プロセスを伴うようなジェネリック・オブジェクト・ネットワークを含んだ仕事を分担する。

【0128】以上述べたことを、インタラクションの形態について説明したオブジェクト指向の属性値についての継承および合成型の観点からシステムを考えると、継承型データ：継承型データをもとにある処理を要求(Request)するエイジェント機能を行うサーバから具体的テンプレートを受け取る。即ちそのデータに対応したテンプレートの型として定まったデータ容器にデータ値が格納された具体的テンプレートを受け取る。この仕事を担当するスペシフィック・ロール・サーバもエクステンシブル WELL によって記述され、実行を行う。具体的実行は自身が行っても、他のサービスモジュールに依頼してもよい。

合成型データ：上記によりRequest されたデータをもとに依頼されたスペシフィック・ロール・サーバが実行処理を行う。その結果を格納すべきデータ型としてのテン

プレート型式は、Request 時に受け取っており、それに格納して応答としてエイジェント・ロール・サーバに返す。この一連の仕事もエイジェント・ロール・サーバとの間に存在するコモン・プラットフォーム(これはスペシフィック・ロール・サーバがもっている)とそれを支援するコミュニケーション・マネージャが行う。

【0129】以上のエイジェント・ロール・サーバ3' と、スペシフィック・ロール・サーバ3" との関係は、クライアントとServerとの間のエクステンシブル WELL の処理形態と全く同様である。しかしながらエイジェント・ロール・サーバが複数のデータをもとに継承型データとして複数のスペシフィック・ロール・サーバに処理作業を依頼した時には、制約処理をエイジェント・ロール・サーバが行わなければならない。例として前述したTexture picture の描画を考えるとFlow linesとCell pictureとの描画を、それぞれ独立のスペシフィック・ロール・サーバに依頼し、その結果が返された時、Flow linesとCell pictureとの間の制約条件を満足させなければならないので、そのための制約条項が図25のIntegratingConstraints についての属性値とSatisfying文によつての規定とを満足させる。

【0130】この制約処理をコミュニケーション・マネージャに行わせる。したがって、コミュニケーション・マネージャは、

(1) テンプレート構造の役割に応じた変換を行う。

(上記表1参照)

(2) テンプレート内容および複数のテンプレート内容についての制約処理に関連した整合化のための通信管理作業を行う。

【0131】この制約処理は形態制約である。それはFlow linesとCell pictureとの2個のサービス・オブジェクト相互に関連していることによる。このため、図27のように制約処理の間での統合がはかられる。

【0132】図27において、

① クライアント1は、ウィンドウのデータを眺めながら、オブジェクト・ネットワークによって要求を発する。

【0133】② エイジェント・ロール・サーバ3' は、サービス計画用のデータを用意し、スペシフィック・ロール・サーバ3" に対して、即ち所望する処理を行うための各 Flow line 処理用のサーバや Cell picture 処理用のサーバや Texture picture 処理用のサーバに対して、要求を発しおよび/または処理の継承を指示する。

【0134】③ スペシフィック・ロール・サーバ3" における各サーバは、夫々自己の処理を行い、処理結果にもとづいた反応を返しおよび/または処理を合成して、エイジェント・ロール・サーバ3' へ応答する。

【0135】④ エイジェント・ロール・サーバ3' は、サービス内容をチェックし、制約条件があるか否か

や現在の制約条件に追加や変更があるかなどを、クライアント1に尋ねるように処理する。

【0136】このように図27に示す如く、スペシフィック・ロール・サーバ3'中の各サーバが処理した結果あるいは所定の制約条件の下で処理した結果について、改めて、各処理結果の間で生じうる制約条件を新たに考慮してゆくようにする。即ち、制約処理の間での統合がはかられる。

【0137】図27に示すプロセスによって図25で述べたAgent roleとSpecific roleとの差異が示されている。Flow lineとCell pictureとについてもFlowとCellとの修飾に関連した制約処理はagent roleの働きと考え、要素ネットワークに対応する処理はspecific roleの働きによって行われる。

【0138】また画像処理と画像描画とを相互作用のある形で、コモン・プラットフォームを用いて対話的情報処理を行うことができる。例えば、エッジ線やハイライト線のような特徴抽出を行う画像処理を行い、抽出されたハイライト線を移動させる。すなわち、左目から見た左画像が与えられた時に視点を右目の位置に移動させて右画像を求めるようにして、立体画像を得たり、あるいは顔の目尻を上下させて表情変化をさせる時に目尻の位置を検出するような画像処理によって表情の特徴を得たりすることができる。

【0139】しかし、これら特徴検出については、サーバがいつも完全な実行処理できるとは限らない。そこで明確な特徴を検出する箇所をクライアントが選び、そこである性質としての特徴名としての制約を示して特徴を表現する点の候補を求め、次いでその中から特徴点を確定するようにする。この操作は例えば図8の要素ネットワークにおいて、クライアント側の動作を時系列的にC1, C2...として示し、システム側の処理を時系列的にS1, S2...として示すと次の如く処理が進行される。即ち、

(C1-1) Colored Region Segmentの項目名を指定し、データ・ウィンドウ上で抽出したい特徴の明らかな、

(C1-2) Region SegmentをIdentifyする。

(C1-3) ついで、Featuredをキーボードから入力し、Colored Pointの項目を指定することにより、Featured Colored Pointを求めたことを、クライアントの意図として、システムに要求する。

(S1) システムはメッセージ・ウィンドウによって意図を確認してSeek featured Colored Pointを表示し、IdentifyされたColored Region Segmentを表示する。

(C2) Featuredの制約としてHighlightをキーボードから入力する。Highlightの性質はシステムに既に与えてあるとする。

(S2) Featured(Highlight) Colored Pointを抽出する。

(C3~S3) Featured(Highlight)の単語をコピーし、Colored Line SegmentをIdentifyする。これにより、システムは、クライアントがFeatured Colored Point Sequenceを経てFeatured Colored Line Segmentを作るべきことを、サービスとして実行する。

【0140】以上によって、Highlight線を構成し、クライアントはその確認を行った後、それを移動した上で新Highlight線およびエッジ線を左右の眼の視差に対応して移動する。この際、Colored属性を継承し、その上でHighlight線の特徴を維持するように描画サービスがシステムが行う。このHighlight線の移動差をmoved highlight線として修飾を行い、moving vector diagramを用いて時間的な移動を行うようにする。

【0141】またこれらの修飾を行うのに当たって、colored...によって輝度や色度の属性用テンプレートを付加したのと同様に追加テンプレート部分が付加される。これらは名詞オブジェクトに対するものであったが、動詞オブジェクトについてもpointからline segmentに至るdrawが継承される。したがって動詞オブジェクトについても実行開始時、実行中、実行終了時の制約条件が付加される。

【0142】また実行要求と制約との関係としては、サービスとしての実行要求を行いうる条件を規定することが必要で、

(1) 属性値処理のために、エイジェント・ロール・サーバ3'からの継承性属性のみが要求により実行可能で、結果は合成属性として上位エイジェント・ロール・サーバ3'へ応答するようにされる。

(2) 実行要求は、クライアントによってオブジェクト・ネットワーク上の動詞オブジェクトを指定し、もしくは上述のような実行の単位(pointからlineへのdraw)によって実行系列を定めるようにして、委任発行される。

(3) ジェネリック・オブジェクト・ネットワーク上のジェネリックな動詞オブジェクトについても実行系列命令の生成を(2)と同様に行う。

【0143】このために図4に関連して記述した時相的制約が満足されていることを制約処理によって調べる。実行結果をエイジェント・ロール・サーバ3'に回答すると、エイジェント・ロール・サーバ3'は制約処理のプロセスに入る。制約条件を満足していない時には、合成データを応答してきたスペシフィック・ロール・サーバへ継承属性を変更させて再実行に移らせる。それでも不満足な時には合成ルートをたどって(図22、図23の中継などによって)最終的にクライアントの判断を仰ぐパスが設定される。

【0144】これまでの説明は、クライアントが一人であったが、複数のクライアントの協調化も行いうる。それはクライアントの上位階層として協調化を行う協調化エイジェントが存在すると考えることによって、コモン・プラットフォームを協調化して実行する場として使用

可能である。

【0145】グラフ構造化エディタは、オブジェクト・ネットワークの定義を読み取り、それを可視化可能なオブジェクト・ネットワークに表示すると共に、各オブジェクトのテンプレート構造および制約条件を各オブジェクトの属性として管理している。したがってエクステンシブル WELL の機能によってエイジェント・ロール・サーバ、スペシフィック・ロール・サーバのソフトウェアが記述される。そしてオブジェクト・ネットワークとテンプレートの構造とは、システム設計の最初の段階で、開発担当のその分野を担当するエキスパートによって記述が行われる。

【0146】サービス・モジュールが、できるだけ基本的なソフトウェア・モジュールとそれに対応する専門分野との組み合わせという考え方で、開発し用意される。並行処理のためには、オブジェクト・ネットワークの構造を図28のように、TOP を目的ゴール、BOTTOMを出発点として表現する。当該BOTTOMは上述の図8のNONEの如きものと考えてよい。

【0147】図28においては、現在実行中のタスクが今CとEとであるとすると次に並行実行可能なタスクがAとDとであり、AとDとの並行実行が終了した時には次にAとBとに移ることを可能とすることが示されている。このように、オブジェクト・ネットワークの構造を解析して、並行実行の進行と共に複数のサーバの並行実行を動的に制御させるようにする。

【0148】図2において、クライアント1とエイジェント・ロール・サーバ3' とスペシフィック・ロール・サーバ3'' との関係を示し、クライアント1とエイジェントとの間に表示的プロセス2-1としてのコモン・プラットフォーム2が存在し、かつエイジェントと各サーバ#i との間に従属表示的プロセス2-2としてのコモン・プラットフォーム2が存在する態様を示した。

【0149】図2においてクライアント1の存在を示しているが、当該クライアント1として、エイジェント・ロール・サーバ3' に対して処理を希望するユーザを考慮することは言うまでもないが、当該ユーザとエイジェント・ロール・サーバ3' との間の情報交換を援助するエキスパートを更に考慮することができる。

【0150】図29はエキスパートの存在を考慮したシステム構成を示す。図中の符号1, 2-1, 2-2, 3', 3'' は夫々図1や図2に対応している。そして、符号1-1は図1や図2に示すクライアントに相当するユーザ、1-2はエイジェント・エキスパートであってユーザ1-1とエイジェント・ロール・サーバ3' との間の情報交換を援助するもの、1-3はスペシフィック・エキスパートであってエイジェント・ロール・サーバ3' とスペシフィック・ロール・サーバ3'' との間の情報交換を援助するものを表している。

【0151】図示のユーザ1-1は人間であるが、エイ

ジェント・エキスパート1-2やスペシフィック・エキスパート1-3は夫々人間に限られるものではなくインテリジェント機能をもつ処理ユニットであってもよい。

【0152】図29に示すように、エイジェントに対してユーザ1-1としてのクライアントは、ある局所的な特定の問題解決(problem resolution)を依頼する。すなわち、ある問題についてエイジェントの支援をたのみ満足する解を得る。

【0153】これに対し、専門家としてのエイジェント・エキスパート1-2はエイジェントが担当する問題解決を行うための各種サービスの仲介実行を行う仕事を担当する。すなわち、ジェネリック・オブジェクト・ネットワークを構成し、それをエイジェント・ロール・サーバ3' に対して実際に実行するオブジェクト・ネットワーク群に分解し、図22, 図23のような実行を可能とする方法を設計する。

【0154】この際、ジェネリック・オブジェクト・ネットワークとそのためのテンプレート（これにはデータのみでなくサービスモジュール名を含む）と制約条項とをもっている。図22, 図23に示したエイジェントの中継機能は、エイジェント・ロール・サーバ3' のもつウエル・システム11の中核部に備えられている。制約条項の処理は、一般的形式で記述されている部分については制約処理サーバに依頼する。

【0155】ジェネリック・オブジェクト・ネットワークとそれに関連するテンプレートをエイジェント・ロール・サーバ3' のウエル・システム11の言語処理部に記憶し、それを抽象化サービスとして実現させるためには、ウエル・システム11のグラフ構造化エディタの機能を用いてエイジェント・エキスパート1-2が行う。

【0156】同様に、個々のスペシフィック・ロール・サーバ3'' にも、対応するスペシフィック・エキスパート1-3が存在し、この場合のユーザに該当するエイジェントが分配したある特定のサービスを実現するための実際に実行するオブジェクト・ネットワークとそれに関連したテンプレートを設計し、それぞれのウエル・システム11の言語処理に格納する。

【0157】以上のようなジェネリック・オブジェクト・ネットワークおよび実際に実行するオブジェクト・ネットワークはウエル・システム11の拡張機能を利用して動的変更を行うことが可能である。

【0158】同様のことが制約条項についても行われる。すなわち制約条項をその時点での入力データと見なし、制約条項を拡張し、設計を行うような制約処理用オブジェクト・ネットワークをジェネリック・オブジェクト・ネットワークと実際に実行するオブジェクト・ネットワークとの両者を設計し、それによって制約条項変更サービスを行うようにシステムを動的に変化させる。これもそれぞれのエキスパートが計画し、方法を実現させる。

【0159】以上のことは、表1に示したサービス機能の計画設計に関しており、エキスパートの役割となり、エキスパートは、ユーザ1-1とサーバとの間の問題解決手段を提供し、その評価によって動的設計を行う。

【0160】データに関連する部分については、ユーザのもつ得意機能とサーバのもつ得意機能に対応して、それぞれデータを理解し参照指示する機能にもとづいて記述内容のもつ役割を分化させ、全体システムを効率化するようにさせている。

【0161】上述の如く、ユーザ1-1は或る目的意図の下でシステムを使用して成果を得る役目を果たすものであり、これに対してエキスパート1-2（又は1-3）はシステムを計画し設計し実現させてユーザ1-1の使用のために提供する。当該エキスパート1-2（又は1-3）も、エイジェント・ロール・サーバ3' やスペシフィック・ロール・サーバ3" との間でのインタラクションを行う場としてコモン・プラットフォーム2が用いられる。

【0162】ユーザ1-1は可視型のオブジェクト・ネットワークを用いデータを入力するなどして操作を行う。オブジェクト・ネットワークは、オブジェクトとして名詞型オブジェクトや動詞型オブジェクトを用いる。他動詞型のオブジェクトでは目的語データや補語データについても可視型として表現され操作される。名詞や動詞についての修飾語ではテキスト形式の文や語を用いることが許される。上記のウエル・システム11（図3参照）はこの考え方にもとづいて実現されていると考えてよい。

【0163】エキスパート1-2（又は1-3）は、ユーザ1-1が利用できるように、ウエル・システム11の機能を用いてオブジェクト・ネットワークを設定し、名詞オブジェクトについてはデータ構造としてのテンプレートを用意するようにし、動詞オブジェクトについても同様にサービス名や制約処理のためのテンプレートを用意するようにする。またオブジェクト間に制約条項が存在する場合には、テンプレート内に制約条項へのポイントが設定される。このためにグラフ構造エディタが上記ウエル・システム11における言語処理系に設けられてオブジェクト・ネットワークの変更や拡張が動的に行われる。この動的な処理は、ユーザ1-1の意図を実現する上での、キーワードにもとづいたデータベースに対する事項検索によって行われる。あるいは、エイジェント・ロール・サーバ3' の助けをかりてユーザ1-1自身がオブジェクト・ネットワークの書き下しや訂正を行う。勿論並行処理の下で行うこともできる。これらのサービス計画は、エイジェント・エキスパートによって設計され、ジェネリック・オブジェクト・ネットワークとしてコモン・プラットフォーム上に表示される。

【0164】エイジェント・ロール・サーバ3' は、サービス計画として、ユーザ1-1の要求に対応したオブ

ジェクト・ネットワークをユーザに提示し、その結果にもとづいてサービス役割の割当てを行い、個々のスペシフィック・ロール・サーバにサービス仕様を与えると共に、プロトコル仕様にもとづいて具体的に各割当てサービスを並行的に実行させる。

【0165】

【発明の効果】以上説明した如く、本発明によれば、クライアントとサーバとの間に、可視的な表示をもってクライアントに通知しかつクライアントからの意図を受け取るコモン・プラットフォームがもうけられ、コミュニケーション・マネージャがクライアント側への可視的表示を用いた情報授受とサーバ側へのデータスキーマに則したデータ構造での情報授受とを仲介するようにしており、クライアントはディスプレイのウィンドウに表示された画像と対話を行いながら、所望するオブジェクト・ネットワーク例えばカラー画像生成・色付け処理分野に対する処理を実行することが可能となる。

【図面の簡単な説明】

【図1】本発明の原理構成図を示す。

【図2】エイジェントをもつ構成を示す。

【図3】ウエル・システムとオブジェクト・ネットワークとの関係を説明する図である。

【図4】処理の整合を行う状況を概念的に示した図である。

【図5】処理フロー（I）を示す。

【図6】処理フロー（II）を示す。

【図7】コミュニケーション・マネージャの働きを示す図である。

【図8】要素ネットワークの関係を示す図である。

【図9】コモン・プラットフォームをもつシステムにおいてカラー画像生成・色付け処理を行う態様を示している。

【図10】テンプレートのフォーマットを示す。

【図11】コミュニケーション・マネージャの動作態様を示す。

【図12】クライアントの指示に対応する動作を説明する図である。

【図13】情報の交換動作を説明する図である。

【図14】チェック・モジュールの動作を説明する図である。

【図15】制約条件の表示態様を示す図である。

【図16】アイデンティファイ作業を説明する図である。

【図17】FLOW LINESの描画態様を示す図である。

【図18】うろこパターンの画像を表す図である。

【図19】ライン用のテンプレートのフォーマットを表す図である。

【図20】LINESを限定する形容詞節を表す図である。

【図21】属性にFeature 名を記述されたテンプレートを表す図である。

【図22】テキストチャ描画の流れ(1)を表す図である。

【図23】テキストチャ描画の流れ(11)を表す図である。

【図24】オブジェクト・ネットワークに概念を説明する図である。

【図25】サーバ3' とサーバ3'' との処理例を表す図である。

【図26】中心点を表示した図である。

【図27】制約処理の統合を表す図である。

【図28】並列処理を分析した図である。

【図29】エキスパートの存在を表すシステムの図である。

【図30】クライアント・サーバ間の対話モデルを表す図である。

【符号の説明】

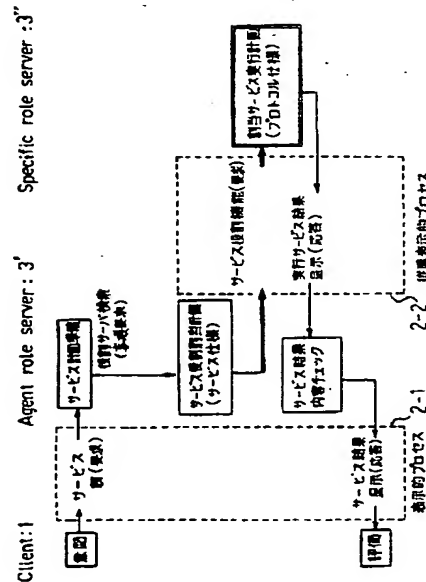
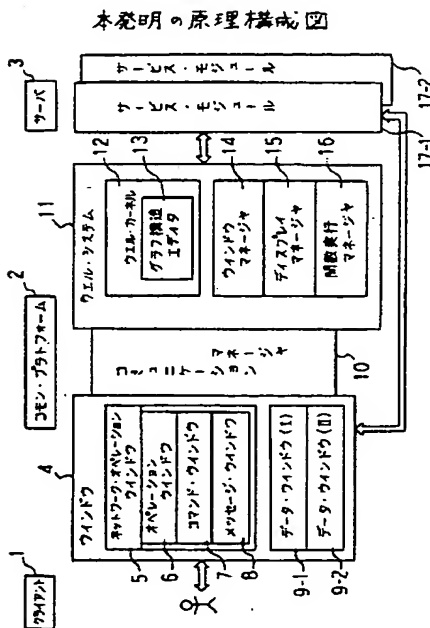
- 1 クライアント
- 1-1 ユーザ
- 1-2 エージェント・エキスパート
- 1-3 スペシフィック・エキスパート
- 2 コモン・プラットフォーム
- 3 サーバ
- 4 ウィンドウ
- 5 ネットワーク・オペレーション・ウィンドウ
- 6 オペレーション・ウィンドウ

【図1】

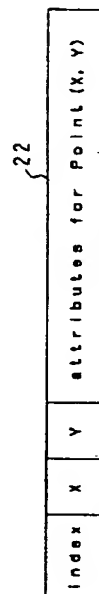
- 7 コマンド・ウィンドウ
- 8 メッセージ・ウィンドウ
- 9 データ・ウィンドウ
- 10 コミュニケーション・マネージャ
- 11 ウェル・システム
- 12 ウェル・カーネル
- 13 グラフ構造エディタ
- 14 ウィンドウ・マネージャ
- 15 ディスプレイ・マネージャ
- 16 関数実行・マネージャ
- 17 サービス・モジュール
- 18 分野
- 18C カラー画像生成・色付け処理分野
- 19 オブジェクト・ネットワーク
- 20 輪郭部 (Frame Section)
- 21 カラー部 (Color Section)
- 22 テンプレート
- 23 バッファ
- 23' バッファの内容
- 20 24 チェック・モジュール
- 25 制約条件
- 27 名詞オブジェクト
- 33 ジェネリック・オブジェクト・ネットワーク
- 34 特定のパラメータや制約条件
- 35 スペシフィック・オブジェクト・ネットワーク

【図2】

【図10】

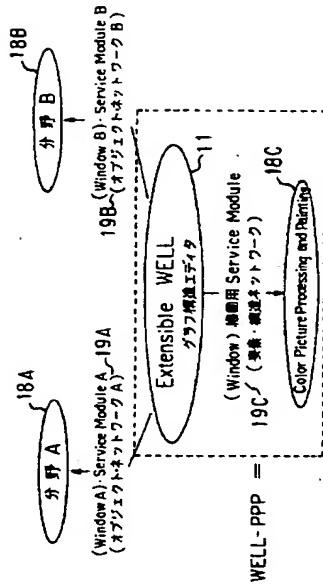


テンプレート



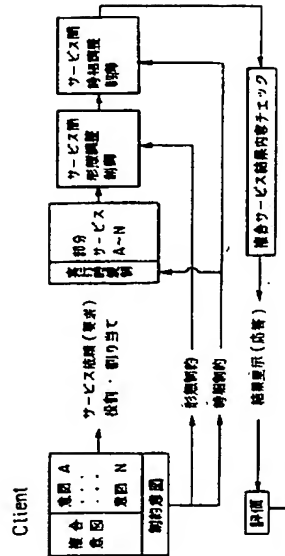
【図3】

ウェルとオブジェクトネットワーク



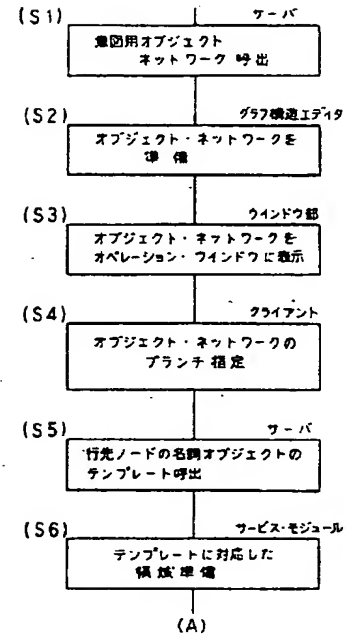
【図4】

処理の整合



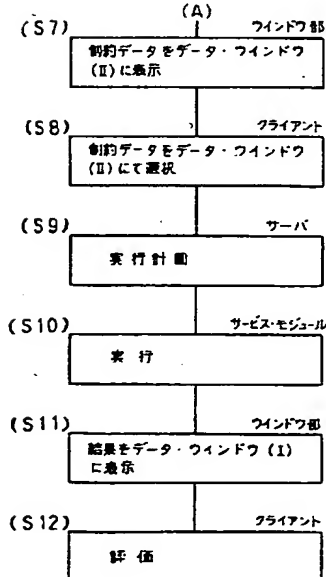
【図5】

処理フロー(1)



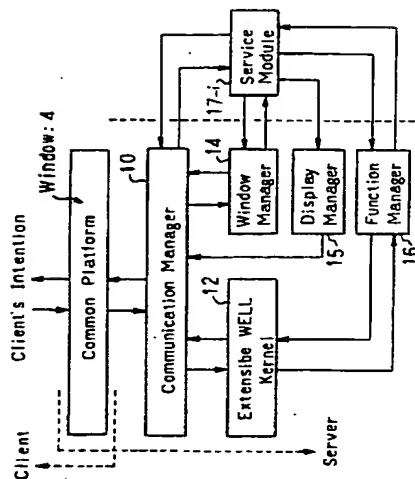
【図6】

処理フロー(II)



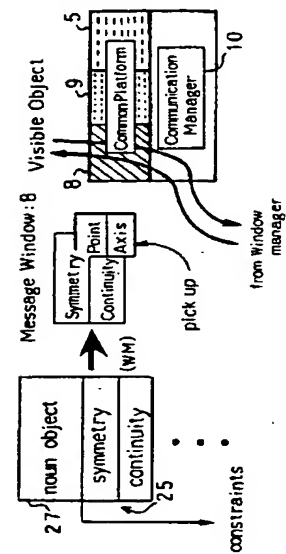
【図7】

コミュニケーション・マネージャの働き

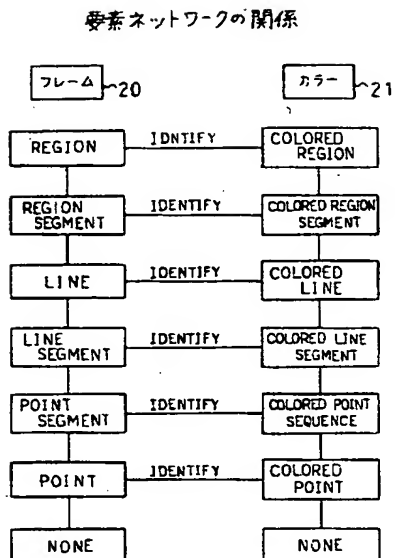


【図15】

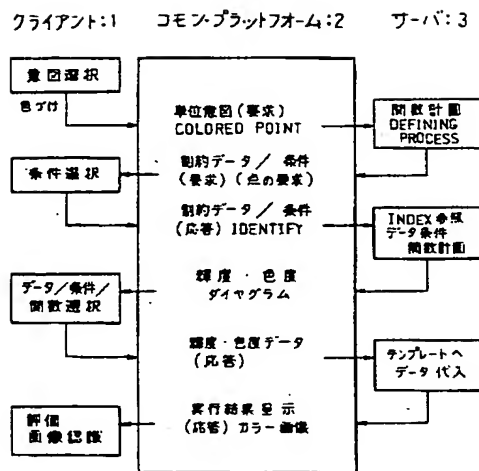
制約条件の表示



【图 8】



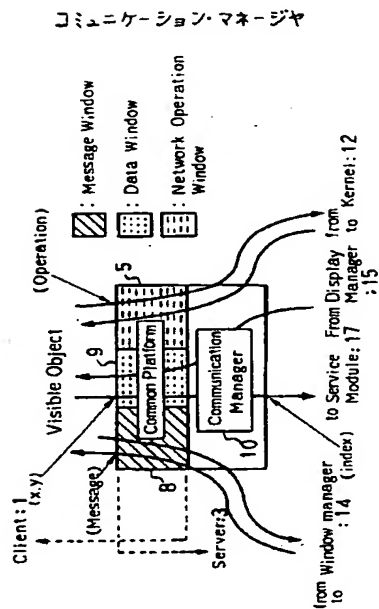
【图9】



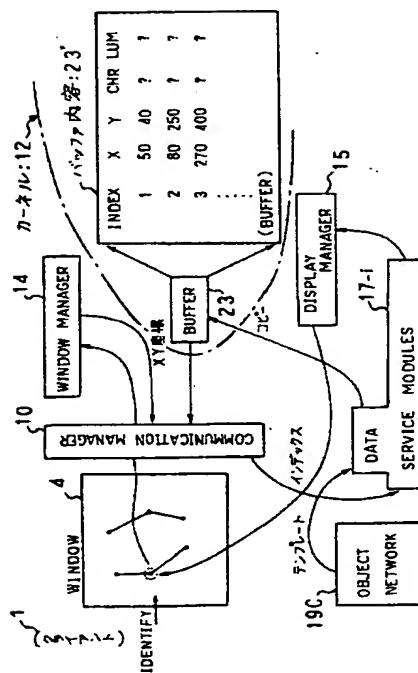
【图 1 2】

【图 18】

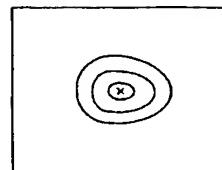
【图 1-1】



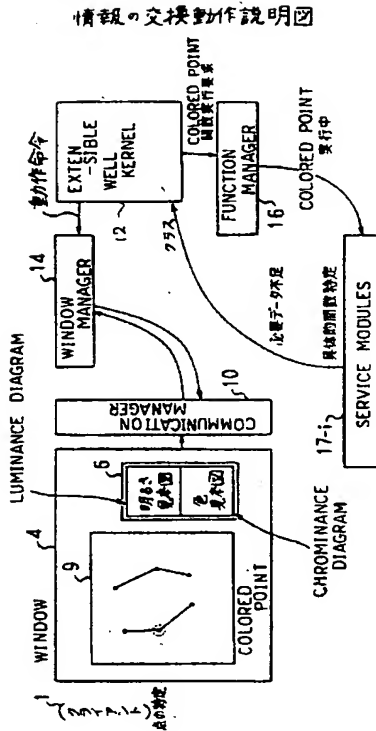
指示に対応する動作説明図



うろこパターンの画像



【図13】



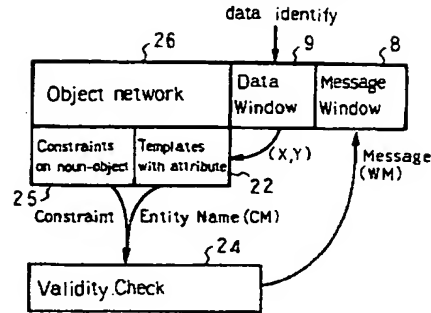
【図21】

テンプレート

緯度	色度ベクトル	Feature名
----	--------	----------

【図14】

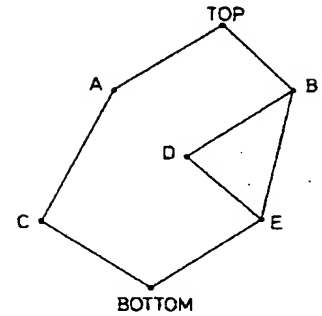
チェック・モジュールの動作



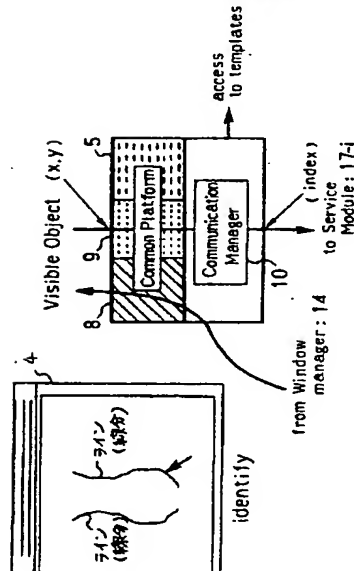
【図16】

【図28】

並行処理の分析

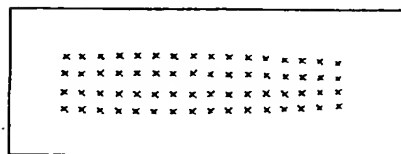


アイデンティファイ作業

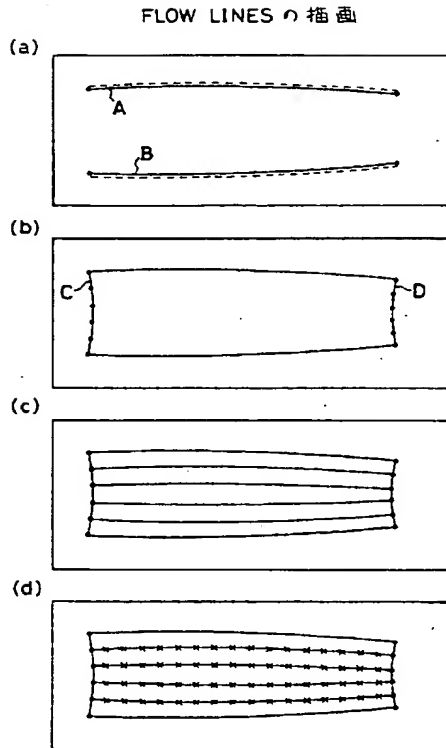


【図26】

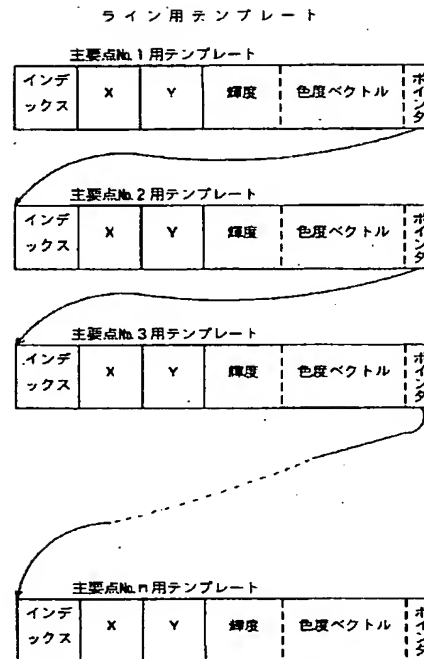
中に点の表示



【図17】

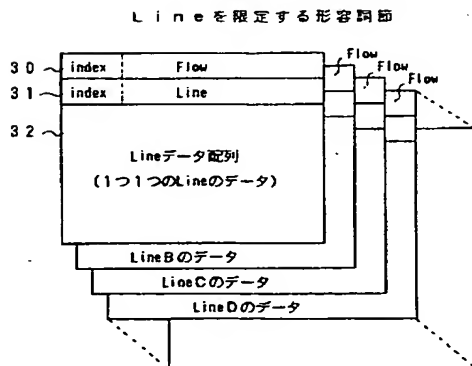


【図19】

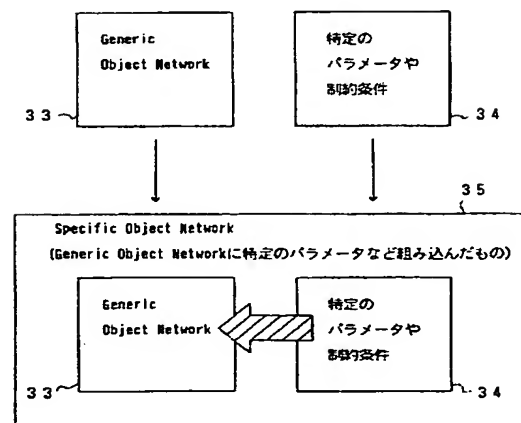


【図24】

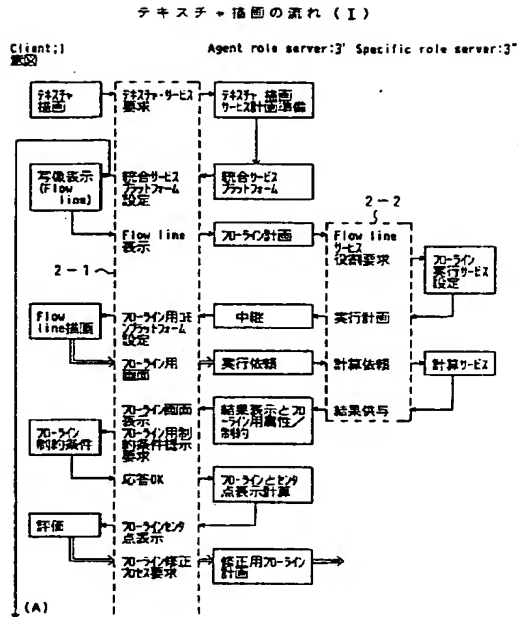
【図20】



オブジェクト・ネットワーク

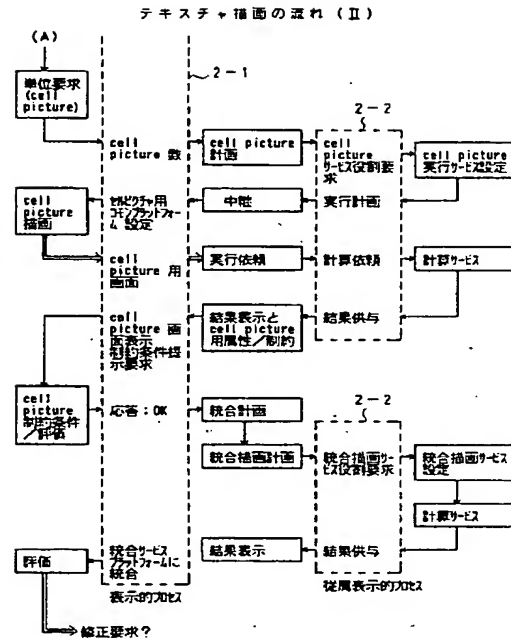


【图 2 2】



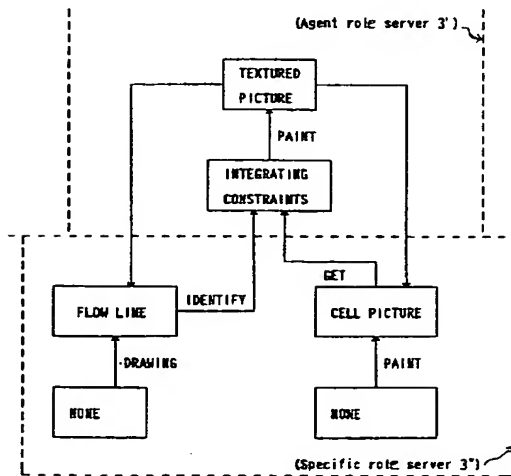
【図 25】

【图 23】

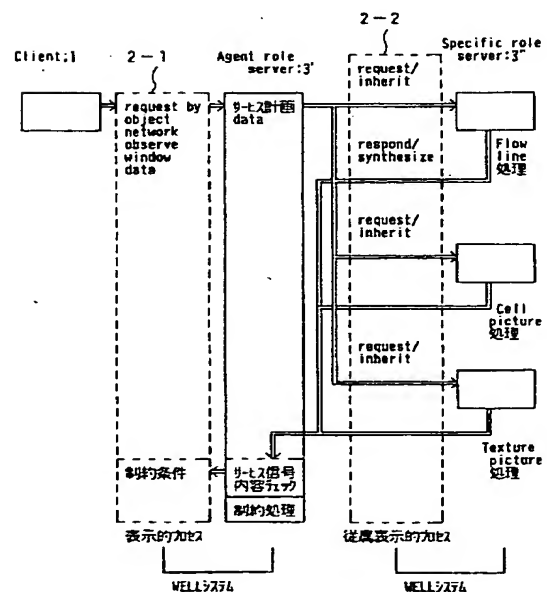


【图 2 7】

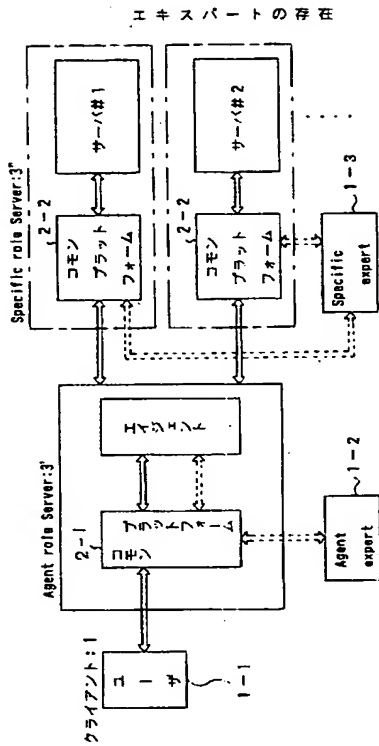
サーバ 3' とサーバ 3'' との処理例



制約処理の統合



【図29】



【図30】

